Titre: Title:	Étude du potentiel de création de formes du quantron
Auteur: Author:	Julien Hackenbeck-Lambert
Date:	2011
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Hackenbeck-Lambert, J. (2011). Étude du potentiel de création de formes du quantron [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/623/

UNIVERSITÉ DE MONTRÉAL

Étude du potentiel de création de formes du quantron

JULIEN HACKENBECK-LAMBERT
DÉPARTEMENT DE MATHÉMATIQUES ET GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MATHÉMATIQUES APPLIQUÉES)
AOÛT 2011

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

Étude du potentiel de création de formes du quantron

présenté par : HACKENBECK-LAMBERT, Julien
en vue de l'obtention du diplôme de : Maîtrise ès Sciences Appliquées
a été dûment accepté par le jury d'examen constitué de :

M. CLÉMENT, Bernard, Ph. D., président
M. LABIB, Richard, Ph. D., membre et directeur de recherche
M. ADJENGUE, Luc-Désiré, Ph. D., membre

REMERCIEMENTS

À mes parents Claude et Carole, pour leur support constant et leurs nombreux encouragements, je tiens à leur exprimer mes plus sincères remerciements.

Il en va de même pour mon directeur Richard, sans l'aide de qui ce travail n'aurait pas été possible. Merci pour sa disponibilité, sa flexibilité et son expertise.

RÉSUMÉ

Ce travail consiste en l'étude du potentiel de création de formes du quantron, un modèle récent de neurone artificiel qui décrit la transmission de l'information aux synapses des neurones à l'aide de processus stochastiques. Celui-ci a auparavant été le sujet de différentes tentatives d'approximations mathématiques dans le but d'applications futures au domaine des réseaux de neurones. Nous proposons plutôt une implémentation informatique fidèle au modèle original qui nous permet d'étudier ses caractéristiques. Dans le cadre de la reconnaissance de formes, nous explorons les capacités du quantron et de réseaux formés à partir de celui-ci à créer des images binaires à deux dimensions. Plus spécifiquement, nous nous sommes donnés comme objectif de déterminer les architectures et les paramètres des réseaux qui permettent de produire les 26 lettres majuscules de l'alphabet. D'abord en variant les paramètres de quantrons uniques selon des règles déterminées par notre connaissance préalable du fonctionnement du modèle, nous obtenons une grande quantité d'images que nous arrivons à assembler pour obtenir quelques premières lettres. En procédant à l'analyse des fonctionnements internes des quantrons qui produisent les lettres, nous arrivons à améliorer leur esthétisme et par le fait même acquérons une meilleure compréhension du modèle, ce qui nous mène à découvrir et démontrer des caractéristiques nouvelles lui donnant un puissant potentiel de production d'images. Des limites et contraintes sont aussi mises à jour, nous suggérons par la suite comment elles peuvent être contournées. Le souci de fidélité au modèle original du quantron est conservé tout au long du travail et les effets des approximations inhérentes à l'implémentation numérique sont étudiés. Nous proposons finalement des équations pour déterminer les paramètres de réseaux produisant n'importe quelle image désirée et les utilisons pour compléter la banque de lettres de l'alphabet. Les résultats obtenus suggèrent un avenir prometteur au modèle et différentes avenues sont suggérées pour tirer profit.

ABSTRACT

This document presents the study of pattern creation capacities of the quantron, a recent model of artificial neuron that describes the transmission of information at synapses with stochastic processes. Earlier works have been dedicated to the development of mathematical approximations that aim at the applicability of the model within the neural network domain. Rather, we propose an informatics implementation true to the original mathematical model that enables us to study its characteristics. Within the guidelines of pattern recognition, we explore the capacities of quantrons and of networks of quantrons to produce binary two dimensional images. Specifically, we had as an objective to determine the architectures and parameters of networks able to produce the 26 capital letters of the alphabet. First by varying the parameters of single quantrons within limits and rules determined by our prior understanding of the model, we obtain a great quantity of images which we are able to assemble in order to create a few first letters. By proceeding to the analysis of the inner workings of the quantrons producing the letters, we are able to improve their esthetics and at the same time obtain a better understanding of the model, which leads us to discover and demonstrate different new characteristics that give it strong image production capacities. Limitations are also brought to light for which we suggest ways to work around. In order to support the precision of the numeric approximations of the model, its effects are studied and carefully monitored. Finally, we propose equations that enable us to find parameter values that produce any desired image and make use of them in order to complete our collection of letters of the alphabet. The results suggest a promising future to the model and different avenues are suggested for later work to benefit from it.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

MLQ	Quantron multicouches.
MLP	Perceptron multicouches.
Γ	Seuil d'activation.
θ	Délai du signal.
$\vec{\Theta}$	Vecteur des délais.
S	Largeur des noyaux.
$\vec{S}$	Vecteur des largeurs.
w	Hauteur des noyaux.
$\vec{W}$	Vecteur des hauteurs.
w_{x1}	Lorsqu'il y a plusieurs quantrons, w_{x1} est le paramètre de hauteur des noyaux w pour l'entrée X du 1er quantron. θ_{y2} est le paramètre de délais pour l'entrée Y du 2e quantron etc. Lorsqu'il n'y a qu'un quantron mais plusieurs utilisations de la même entrée, le chiffre indique plutôt à quelle entrée le paramètre appartient.
$w_{x1,1}$	Lorsqu'il y a plusieurs quantrons utilisant plusieurs fois la même entrée, le premier chiffre représente un numéro du quantron et le second chiffre le numéro de l'entrée.
w_{S1}	Lorsqu'il y a plusieurs quantrons et un quantron à la couche de sortie, w_{S1} est le paramètre de hauteur des noyaux w pour l'entrée provenant de la sortie du 1er quantron de la couche d'entrée. θ_{y2} est le paramètre de délais pour l'entrée provenant de la sortie du 2e quantron etc.
n	Nombre d'entrées.
N	Nombre de noyaux.
Ω	Sortie du quantron.

LISTE DES ANNEXES

Chapitre 1 : INTRODUCTION

1 CONTEXTE

« Si le cerveau était assez simple pour que nous puissions le comprendre, nous serions trop simples pour le faire. » — Lyall Watson

1.1 Historique

La principale caractéristique qui distingue l'homme du reste des êtres vivants de la planète est sa capacité à s'interroger sur son existence. Par son intelligence évoluée, il arrive à développer une conscience qui le mène à s'interroger sur la source et la raison d'être de celle-ci. Ces questions ont longtemps été du ressort de l'ésotérisme, de la religion et de la philosophie, les sciences n'étant pas encore assez développées pour tenter d'y apporter leurs propres réponses. Il demeure que l'intelligence est associée au cerveau dès l'Antiquité. Hippocrate (460-370 av. J.-C.) disait : « … je regarde le cerveau comme l'organe ayant le plus de puissance dans l'homme, car il nous est, lorsqu'il se trouve sain, l'interprète des effets que l'air produit; or, l'air lui donne l'intelligence.». Par la suite, la compréhension scientifique du fonctionnement du cerveau ne connait pas de grandes avancées et reste limitée jusqu'à relativement récemment.

C'est vers la fin du 19^{e} siècle qu'est découvert l'élément de base de la construction du cerveau, le neurone. Ces cellules nerveuses traitent l'information provenant des différents sens sous la forme d'impulsions électriques dans un énorme réseau de connexions, dont le nombre est estimé à plus de 60 billions ($6x10^{13}$) (Shepherd & Koch, 1990), et sur lesquelles la biologie du siècle dernier n'a cessé d'en apprendre davantage. Entre-temps, les mathématiciens se sont penchés sur le problème de la reproduction de la cognition humaine et ont donné naissance à un nouveau domaine, l'intelligence artificielle.

En 1936, le mathématicien A. M. Turing établit les bases d'une machine qui, contrairement à une machine à écrire qui n'arrive qu'à produire des symboles, est aussi capable de les interpréter et d'agir en conséquence (Turing, 1936). La machine de Turing est l'ancêtre de l'ordinateur moderne capable de tant d'exploits inimaginables seulement un siècle plus tôt. Évidemment,

l'exemple réel et fonctionnel du cerveau humain ne passe pas inaperçu et certains n'hésitent pas à s'en inspirer.

À l'époque de la Seconde Guerre mondiale, le psychiatre W. S. McCulloch s'associe au mathématicien W. Pitts pour la rédaction d'un article présentant une unité de calcul logique provenant de leur étude neurobiologique (McCulloch & Pitts, 1943). Ils y montrent qu'en principe, une quantité suffisamment grande de ces unités, si placées en réseau, pourrait résoudre n'importe quelle équation. Les réseaux de neurones étaient nés. Quelques années plus tard, F. Rosenblatt propose le perceptron, un modèle mathématique simple du fonctionnement d'un neurone inspiré des travaux de McCulloch et Pitts (Rosenblatt, 1958). Les valeurs d'entrées y sont multipliées par des poids puis sommées. Un biais est ajouté au résultat qui est ensuite traité par une fonction de transfert non linéaire puis transmis à la sortie. Peu de temps après, un algorithme d'apprentissage supervisé par minimisation des erreurs est développé (Widrow & Hoff, 1960). Celui-ci permet alors d'effectuer l'apprentissage d'un problème à un réseau d'une couche de perceptrons en modifiant les poids des entrées.

La percée crée un véritable engouement pour le domaine qui prend par contre abruptement fin neuf ans plus tard avec la parution du livre « Perceptrons » coécrit par M. Minsky et S. Papert dans lequel ils démontrent des limites importantes aux problèmes solutionnables par un réseau de perceptrons d'une seule couche (Minsky & Papert, 1969). Ces limites provoquent un ralentissement des recherches dans le domaine qui dure près de quinze ans.

C'est seulement en 1986 que le trio de D. Rumelhart, G. Hinton et R. Williams fait renaître les réseaux de neurones en introduisant l'algorithme de rétropropagation de l'erreur (Rumelhart, Hinton, & Williams, 1986). Celui-ci permet cette fois l'apprentissage de réseaux de perceptrons multicouches (MLP) qui sont alors capables de passer outre les limitations de leurs ancêtres à couche unique. Ces nouveaux réseaux font leurs preuves dans plusieurs applications et dès lors le domaine connait une explosion de popularité.

En 1999, R. Labib propose le quantron, nouveau modèle de neurone artificiel pouvant être considéré comme hybride entre le perceptron et le neurone à impulsion (Labib, 1999). Celui-ci se

base sur une modélisation stochastique de la transmission des neurotransmetteurs dans les fentes synaptiques qui relient les neurones.

Cette étude se consacre au développement du potentiel du quantron. Nous l'explorons dans le cadre de problèmes de classification dans le plan bidimensionnel afin d'en déduire ses caractéristiques originales ses performances. Les deux prochaines sous-sections résument le modèle et les recherches antérieurs effectués sur celui-ci afin de bien situer le travail présent.

1.2 Le modèle du quantron

Nous utilisons dans ce travail les équations qui définissent le modèle du quantron. Nous exposons celles-ci ici à titre indicatif. Les fondements théoriques qui les supportent sont aussi résumés. Le lecteur peut désirer d'abord survoler cette sous-section et n'y revenir qu'en cas de besoin.

L'objectif du quantron est d'établir un modèle mathématique du neurone qui soit plus fidèle au modèle biologique que l'est par exemple le Perceptron. Il tire son inspiration des avancées en neurobiologie, particulièrement la méthode de transmission de l'information entre les neurones, pour développer son nouveau modèle.

Les neurones communiquent entre eux grâce aux connexions qu'ils forment. C'est à ces connexions, appelées synapses, que l'information est transmise à l'aide de neurotransmetteurs. En fait, à l'intérieur même d'un neurone, l'information se propage par la polarisation et la dépolarisation des membranes tel un signal électrique le long d'un câble. Une fois que ce potentiel d'action atteint la synapse, des neurotransmetteurs sont libérés et diffusent dans l'interstice vers le neurone postsynaptique. Chacune de ces particules qui atteignent les récepteurs de la membrane contribue à l'augmentation de sa charge jusqu'à l'atteinte d'un seuil où un potentiel d'action est déclenché dans le neurone postsynaptique. La figure suivante illustre ce phénomène.

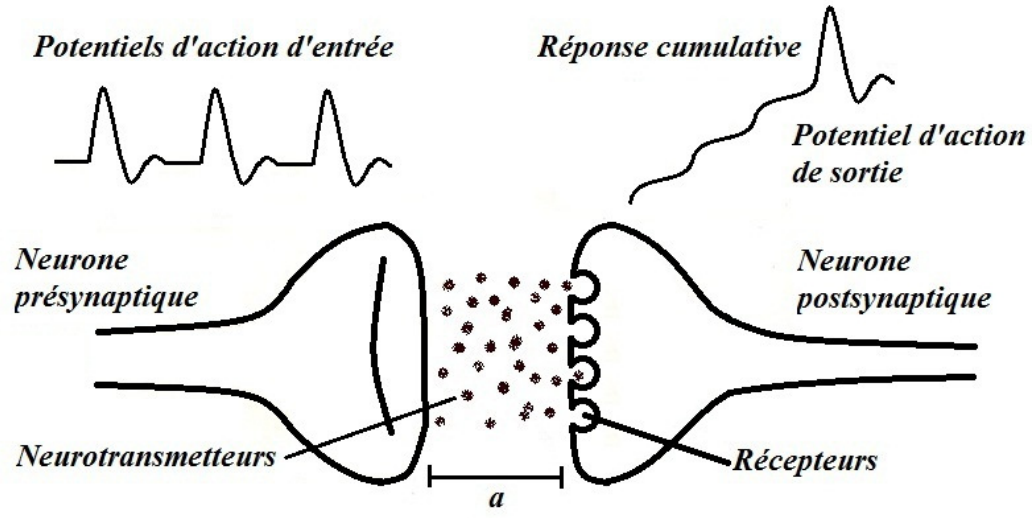

Figure 1.1: Synapse et neurotransmetteurs

En utilisant le mouvement brownien géométrique, on en arrive à une équation décrivant l'accumulation de la charge $V(t)$ sur la membrane postsynaptique en fonction du temps t pour un seul potentiel d'action d'entrée et une fente de largeur a :

$$V(t) = \frac{2w}{\sqrt{2\pi t}} \int_{ln\ a}^{\infty} e^{-x^2/2t} dx = 2w\left(1 - \Phi\left(\frac{\ln a}{\sqrt{t}}\right)\right) \tag{1.1}$$

où w est une constante qui fera partie des paramètres du quantron et est un équivalent aux poids du perceptron. La fonction $\Phi(\cdot)$ est la fonction de répartition de la variable aléatoire normale centrée réduite. Un second paramètre S qui représente la durée de vie de l'association neurotransmetteur-récepteur est ensuite introduit. Après ce temps, la membrane du neurone postsynaptique se décharge et retrouve son état initial de façon symétrique. On obtient la variation finale de polarisation $\varphi(t)$ causée par le potentiel d'action d'entrée :

$$\varphi(t) = V(t) + \big(V(S) - V(t) - V(t - 2S)\big)u(t)$$
$$+\big(V(t - S) - V(S)\big)u(t - 2S) \tag{1.2}$$

où $u(t)$ est la fonction échelon.

C'est avec ces modèles mathématiques que le quantron est construit. En définissant $x_i{}^{-1}$ comme la fréquence d'arrivée des potentiels d'action à différentes synapses du neurone, et donc x_i la période, on obtient la formule finale du signal représentant le potentiel $R(t)$ de la membrane du neurone postsynaptique pour n entrées :

$$R(t) = \sum_{i=1}^{n} \sum_{k=0}^{+\infty} \varphi(t - \theta_i - kx_i) \tag{1.3}$$

où les θ_i sont un troisième paramètre de délai qui représente le temps qui s'est écoulé avant que le premier potentiel d'action ne survienne, ce qui ajoute à la flexibilité du modèle.

Bien que la charge s'accumule sur la membrane selon la formule précédente, l'information n'arrive à être transmise que si cette charge parvient à atteindre une valeur seuil notée Γ. Si c'est le cas, un potentiel d'action est déclenché, sinon rien ne passe. La sortie Ω du quantron est définie comme suit :

$$\Omega = \begin{cases} 0 \; si \; R(t) < \Gamma \; \forall t \in [0, \infty) \\ \alpha \qquad \text{sinon} \end{cases} \tag{1.4}$$

où

$$\alpha = inf\{t > 0 : R(t) = \Gamma\}$$

est le temps minimum requis pour que le potentiel de la membrane postsynaptique atteigne le seuil Γ et déclenche un potentiel d'action. Si ce seuil n'est jamais atteint, α prend la valeur 0.

La valeur de la sortie est définie afin de permettre à un premier quantron de se connecter à l'entré d'un second, pouvant de cette façon former un réseau « MLQ ».

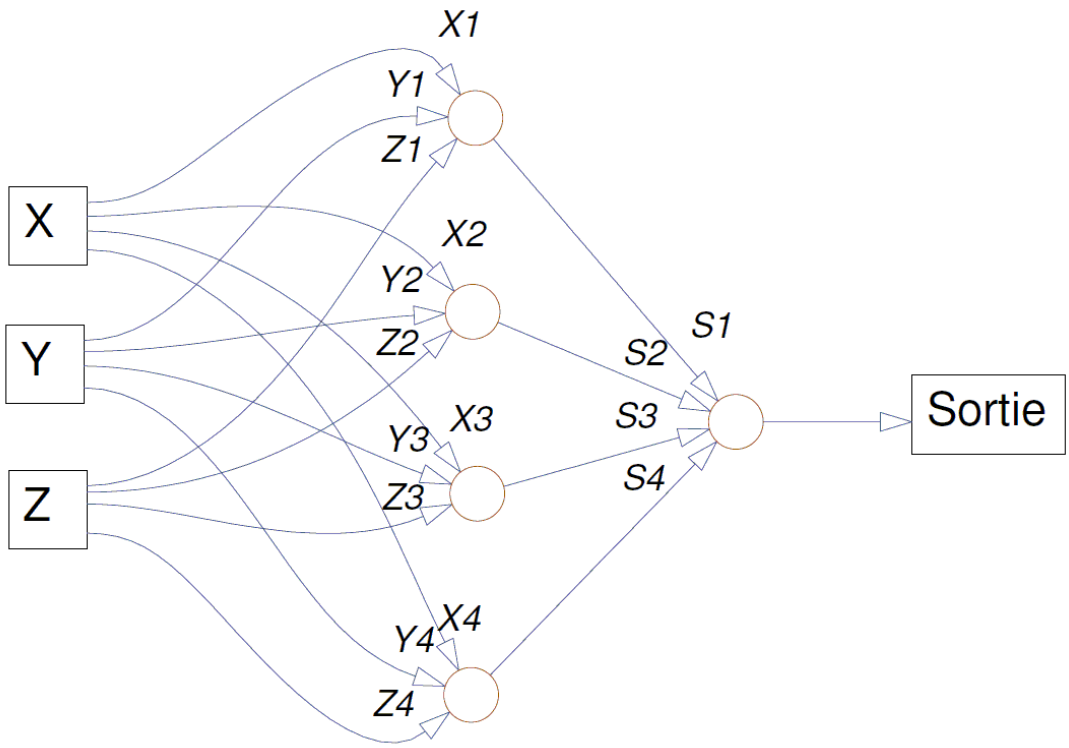

Figure 1.2 : Exemple d'architecture d'un MLQ à 3 entrées, 4 quantrons à la couche d'entrée et 1 quantron de sortie

Il est aussi possible de simplifier cette sortie en la rendant binaire, soit :

$$\Omega = \begin{cases} 0 \; si \; R(t) < \Gamma \; \forall t \in [0, \infty) \\ 1 \qquad \text{sinon.} \end{cases} \tag{1.5}$$

Cette simplification est utilisée pour prouver l'efficacité du quantron en résolvant le problème de classification non linéaire classique du ou exclusif « XOR ». Ce problème consiste en la séparation de 4 points en deux groupes de 2 qui ne peuvent êtres délimités par une simple ligne droite dans le plan.

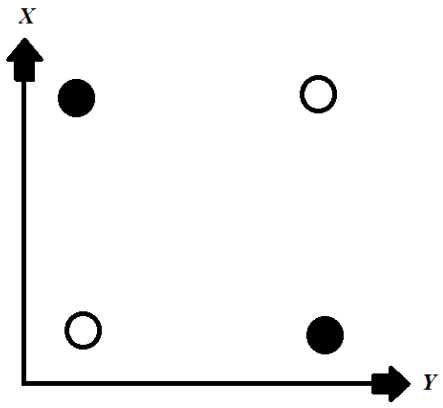

Figure 1.3 : Exemple de classification requise par le problème XOR

Il est montré qu'un quantron à deux entrées possédant seulement six paramètres est suffisant pour bien séparer les classes de ce problème qui requiert au perceptron un minimum de 7 paramètres ainsi qu'une couche cachée (Labib, 1999).

La forme de la courbe de variation de potentiel, ou « noyau », pour l'arrivée d'un seul potentiel d'action est aussi simplifiée à un simple rectangle de telle sorte que :

$$\varphi(t) = w\big(u(t) - u(t - S)\big). \tag{1.6}$$

N'ayant pas de formule analytique pour déterminer précisément le maximum de la fonction d'accumulation de charge nécessaire à la détermination de la sortie, cette simplification permet d'obtenir une courbe plus facile à créer et à observer.

Durant les années suivant sa création, ce nouveau modèle a été la source de recherches dont nous effectuons le survol ci-après.

1.3 Travaux antérieurs

Le principal objectif des recherches effectuées sur le quantron jusqu'à maintenant a été de proposer un algorithme d'apprentissage pour un réseau composé de plusieurs couches : MLQ. L'intérêt des réseaux de neurones étant leur capacité à être modifiés pour s'adapter en fonction des problèmes auxquels ils font face, il est important pour pouvoir utiliser le quantron de découvrir la mécanique qui nous permet de le faire. La source d'inspiration première vient de la

rétropropagation de l'erreur utilisée pour les MLP. L'ennui est que pour être appliqué, cet algorithme nécessite une fonction discriminante continument dérivable, chose vraie pour le perceptron, mais pas pour le quantron. La première étape des recherches est alors de proposer une approximation de la fonction discriminante qui possède cette propriété et c'est surtout sur celle-ci que les auteurs se distinguent.

R. Lastère démontre certains résultats en avril 2005 (Lastere, 2005). Il est confronté à une première difficulté qui consiste en la gérance de la conduction ou la non-conduction du quantron en proposant un algorithme en quatre cas : l'absence de modification, l'excitation du réseau, l'inhibition du réseau et la rétropropagation de l'erreur. La méthode est d'abord testée sur un modèle simplifié du quantron avec des résultats prometteurs. Afin de l'appliquer au modèle standard du quantron, il utilise un noyau rectangulaire et suggère une approximation de la réponse totale par séries de Fourier. À l'aide des termes d'ordre deux du polynôme obtenu, il effectue la rétropropagation pour différents problèmes de classification. Il termine en proposant l'utilisation du noyau original du quantron, l'ajout de techniques heuristiques d'apprentissage et l'élaboration de réseaux mixtes de perceptrons et quantrons exploitant les avantages de chaque modèle.

Peu de temps après, en août, J.-F. Connolly s'inspire des analyses multirésolutions pour proposer une méthode d'approximation multiéchelle de la réponse totale utilisant des paraboles positives à support compact (J.-F. Connolly, 2005; J. F. Connolly & Labib, 2009). Il arrive par la méthode de Laplace à déterminer les maximums globaux de la fonction $R(t)$ qui permettent l'élimination des neurones qui ne transmettent pas d'information. Ses résultats fournissent une fonction dérivable pouvant être utilisée pour l'apprentissage d'un réseau, mais n'en teste pas les performances. Il termine en suggérant l'utilisation d'ondelettes pour la résolution de la fonction du quantron.

En avril 2007, S. de Montigny pousse plus en profondeur l'étude de la performance d'apprentissage à des problèmes simples de classification (De Montigny, 2007). D'abord pour un seul quantron, il utilise une approximation polynomiale pour reproduire les aspects des surfaces discriminantes, mais se limite aux cas où les poids des entrées sont positifs. Il expose certaines

difficultés liées à la complexité des équations mises en cause, mais arrive à des résultats intéressants par la descente du gradient. Les difficultés augmentent lors de la transition au réseau de quantrons pour lequel des modifications à la sortie sont suggérées afin de la rendre continument dérivable. Il termine en suggérant la recherche d'approximations plus complexes pour le quantron et l'étude en profondeur des fonctions d'erreur et des modifications de la sortie du quantron.

En août 2007, J. Pepga Bissou propose aussi un modèle quadratique de la fonction du quantron qui lui permet de construire une architecture de réseau de quantrons multicouches approximés (Pepga Bissou, 2007). Il construit ensuite un algorithme modifié de rétropropagation de l'erreur par la descente du gradient qu'il teste à l'aide d'un exemple d'application de reconnaissance du caractère A défini sur une matrice 5x7. Ces tests démontrent la nécessité de ressources matérielles élevées par rapport au perceptron, réduisant la performance. Il étudie alors la possibilité de l'utilisation du quantron comme filtre de sortie de la machine à état liquide et termine en suggérant cette nouvelle approche comme solution plausible pour résoudre des problèmes de régression.

Plus récemment, P.-Y. L'Espérance prend une voie différente et pousse plus en profondeur le modèle mathématique définissant la forme du noyau du quantron en fonction des considérations biologiques (L'Espérance, 2010). Il utilise différents processus stochastiques pour modéliser les étapes de sécrétion, de diffusion et de réception des neurotransmetteurs et obtient une grande quantité de paramètres. En éliminant les paramètres ayant une influence statistiquement moins significative sur la courbe selon une solution numérique il arrive à une équation analytique. Les nouvelles courbes ainsi obtenues sont comparées aux anciennes et à d'autres mesurées dans d'autres études. Il termine en proposant d'autres éléments biologiques qui pourraient être pris en compte pour l'amélioration du modèle.

Nous voyons qu'à l'exception de la dernière étude, l'ensemble des travaux antérieurs s'attarde à l'approximation de la fonction du quantron dans le but de permettre l'élaboration d'un algorithme d'apprentissage pour un réseau de quantrons multicouche. Les méthodes d'approximation varient selon les auteurs tout comme les performances des exemples d'apprentissages démontrés. Nous

proposons plutôt dans le cadre de ce travail d'explorer les caractéristiques du modèle sans chercher à effectuer un apprentissage, ce qui n'a pas encore été tenté. Effectivement, peu de recherches ont été consacrées au quantron ce qui explique la quantité limité de références disponibles et soutient l'originalité de ce travail.

1.4 Objectifs et cadre du travail

Les réseaux de neurones sont le plus souvent utilisés dans la résolution de problèmes de reconnaissance de formes, c'est donc dans le cadre de ce type d'application que nous proposons de découvrir et d'étudier les caractéristiques et le potentiel jusqu'alors inconnus du modèle du quantron.

La reconnaissance de forme est un domaine où le modèle utilisé, par exemple le réseau de neurones, doit prendre une décision sur l'appartenance d'un objet à une classe parmi un ensemble donné. Un objet est défini par un ensemble de caractéristiques qu'il est possible de quantifier. C'est à l'aide de ces caractéristiques que le modèle doit arriver à décider à quelle classe, ou catégorie, l'objet appartient.

Afin d'effectuer une classification, les caractéristiques de l'objet peuvent être représentées par un seul point dans un espace d'autant de dimensions qu'il y a de caractéristiques. Le problème se résume alors à la détermination de fonctions discriminantes qui définissent des hypersurfaces séparant les régions de l'espace où sont regroupées les classes.

Pour un problème à deux classes, une seule fonction discriminante $g(\vec{x})$ est nécessaire pour effectuer la classification. L'on peut alors, par exemple, attribuer un objet à la classe A si pour ses caractéristiques $g(\vec{x}) > 0$ et à la classe B si au contraire $g(\vec{x}) \leq 0$. Pour le quantron, cette fonction est la simplification binaire de la sortie de l'équation (1.5). Le problème d'apprentissage dit supervisé est alors de déterminer la fonction discriminante qui classifie correctement le maximum d'objets d'un ensemble d'entraînement déjà connu. Plus spécifiquement, dans un espace de deux caractéristiques, les fonctions discriminantes définissent la limite entre deux surfaces d'un plan cartésien. Il est alors possible de représenter ces surfaces sous la forme

d'images où une classe est illustrée par une surface blanche et l'autre par une surface noire. Ceci permet de facilement définir des problèmes à résoudre et de rapidement visualiser les résultats.

Pour un réseau de neurones, la fonction discriminante dépend des paramètres gérant les connexions entre les différentes couches. Le MLP arrive à effectuer l'apprentissage par l'algorithme de rétropropagation de l'erreur. Dans des travaux antérieurs, le MLQ est arrivé par différentes méthodes et approximations à effectuer l'apprentissage de problèmes bien particuliers. Bien que sa capacité à former des surfaces non linéaires a été démontrée (Labib, 1999), rien n'assure que le MLQ soit capable de reproduire n'importe quelle surface demandée. Les travaux s'attardant au problème de l'apprentissage, le problème de consistance n'a pas été abordé.

Le problème théorique de la consistance est très important pour l'apprentissage supervisé d'un réseau de neurones. Il s'agit de déterminer si un réseau est capable de représenter exactement un ensemble d'entraînement particulier, c'est-à-dire de crée une fonction discriminante qui classifie correctement tous les échantillons de celui-ci. Il est en effet d'abord important de s'assurer qu'il est possible pour une entité de « connaître » l'information avant de tenter de la lui faire « apprendre » (Hinton, 2006).

Il a été démontré pour le perceptron que le problème de consistance est résolvable en temps polynomial par programmation linéaire (Megiddo, 1984). Le problème devient NP-complet lorsqu'appliqué au neurone impulsionnel (Maass & Schmitt, 1999). Ceci nous laisse supposer que la preuve de consistance pour le quantron, considérant la similitude avec le neurone impulsionnel combinée à la complexité de sa fonction discriminante, n'est pas un problème simple. Nous proposons tout de même de l'aborder graduellement en explorant la capacité du quantron à produire différentes surfaces discriminantes.

Nous voulons spécifiquement découvrir quel est le potentiel du quantron, particulièrement sa capacité à produire des surfaces discriminantes variées. Nous nous sommes limités aux surfaces de deux dimensions qui peuvent facilement être visualisées sous la forme d'images. Les caractéristiques des images peuvent être étendues aux dimensions d'ordre supérieur. Les images étant numériques, elles sont formées d'une matrice de pixels. Ce sont les coordonnées matricielles de chaque pixel qui sont utilisées en entrée du modèle du quantron. La sortie de

celui-ci détermine la classe du pixel (blanc ou noir) et l'ensemble de ces de l'ensemble de ces classification qu'émerge l'image représentant les surfaces discriminantes.

Afin de fournir une direction à l'exploration des surfaces, nous avons tenté de reproduire les lettres de l'alphabet, plus spécifiquement les majuscules. Ces 26 lettres forment un ensemble de surfaces aux caractéristiques variées : Courbes, diagonales, régions connexes et non connexes, etc.

Une des nouveautés qu'apporte ce travail est que nous n'avons pas tenté d'effectuer d'apprentissage, ce qui nous a permis d'utiliser le modèle mathématique original du quantron sans approximation polynomiale, multiéchelle ou par séries de Fourier. Il demeure qu'afin de produire en temps raisonnable les images nécessaires à l'exploration voulue, il a été nécessaire d'implémenter le modèle informatiquement. Ainsi, la première étape a été de produire une approximation numérique efficace du signal représentant la charge totale $R(t)$.

Afin d'obtenir une bonne représentation des surfaces pouvant être produites par le quantron, un grand nombre d'images ont dû être produites. Compte tenu du nombre de paramètres en jeu et de leurs plages de valeurs possibles, la quantité de combinaisons est théoriquement infinie. De premières relations entre les paramètres ont dû être avancées et des règles déduites afin de borner les plages de valeurs à explorer pour chaque paramètre.

Une fois qu'une banque d'images suffisamment volumineuse a été créée, les lettres ont pu être construites en assemblant les images individuelles par relations logiques entre les pixels. L'amélioration des lettres ainsi produites nous a permis d'étudier les effets des différents paramètres.

Tout au long du travail, nous avons prêté une attention accrue sur toute caractéristique particulière du quantron qui émergeait des observations. Celles-ci ont alors été étudiées plus en détail pour améliorer notre compréhension du modèle.

Chapitre 2 : EXPLORATION

L'exploration présentée dans ce chapitre s'est faite dans le but d'en apprendre plus sur les caractéristiques du quantron et de découvrir ses forces et ses faiblesses. Pour ce faire, nous avons fixé comme objectif de produire des images représentant les 26 lettres majuscules de l'alphabet. Ce but nous a permis au départ de bien orienter notre exploration. Or, au cours de celle-ci, les différentes observations nous ont menés sur des chemins secondaires que nous avons étudiés en parallèle. Les sections de ce chapitre divisent en thèmes le travail effectué durant l'exploration en tentant de respecter le mieux possible l'ordre logique et chronologique de celui-ci. Nous proposons ici un aide visuel à titre de référence pour aider le lecteur à situer les différentes sections. L'ordre chronologique est du haut vers le bas de l'organigramme et les liens logiques sont représentés par des flèches. Nous invitons le lecteur à lui faire référence au besoin.

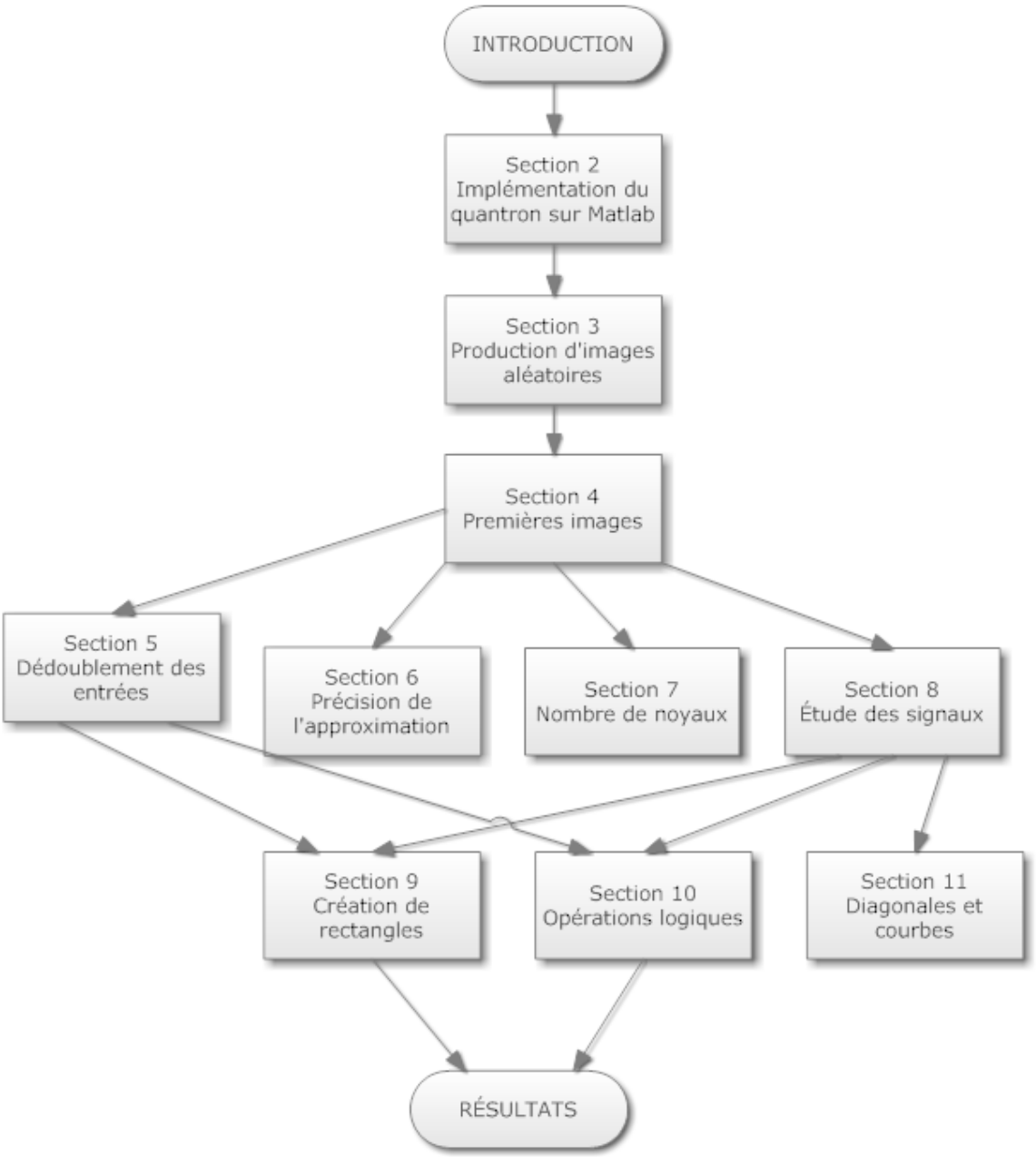

Figure 2.0 : Organisation de l'exploration

2 IMPLÉMENTATION DU QUANTRON SUR MATLAB

Avant de pouvoir commencer à explorer les capacités du quantron, il est bien sûr nécessaire d'implémenter le modèle d'une façon à obtenir les images aussi rapidement que possible. Plutôt que d'utiliser des approximations mathématiques telles que développées lors des travaux antécédents, nous avons développé un modèle informatique. L'objectif de cette transition vers le modèle informatique est de rester aussi fidèle que possible à la définition originale du quantron tout en permettant la création rapide et automatisée d'un grand nombre d'images. L'utilisation d'un logiciel mathématique tel que Matlab implique tout de même une forme d'approximation lors du passage du domaine continu des formules mathématiques reliées au quantron au domaine discret et vectoriel des calculs effectués par le logiciel. Cette approximation impose certaines conditions sur les paramètres et ajoute une préoccupation de résolution de l'approximation utilisée. Ces éléments seront explicités et étudiés en détail dans des sections ultérieures.

2.1 Maximisation de la performance

Nous avons d'abord essayé une approche brute et intuitive qui déterminait itérativement les valeurs de chacun des points du vecteur approximant la réponse totale $R(t)$. Cette approche impliquait le calcul indépendant de chaque valeur et l'utilisation de boucles itératives et de relations logiques peu efficaces en temps de traitement. Le résultat, bien que satisfaisant du point de vue des valeurs obtenues, était très lent à produire une réponse, soit environ une seconde. Considérant le besoin par exemple de 100 réponses pour une image de 10x10 pixels et le désir de produire une quantité d'images dans l'ordre du millier, cette performance était inacceptable. Nous en avons conclu, comme notre expérience préalable avec Matlab le suggérait, que nous devions utiliser des outils de calcul déjà implémentés et optimisés par les créateurs du logiciel afin d'améliorer la performance.

Une observation sur la forme de la réponse totale $R(t)$ vient justement permettre d'utiliser un tel outil. Prenons la définition de la réponse totale pour une des entrées du quantron :

$$R_{x_1}(t) = \sum_{i=0}^{\infty} w_1 \varphi_1(t - \theta_1 - i x_1). \tag{2.1}$$

Nous voyons qu'il est possible de décomposer celle-ci en une convolution entre un peigne de Dirac $P_{x_1}(t)$ défini par les valeurs x_1 de l'entrée et du paramètre θ, et l'équation du noyau défini par sa forme choisie et les paramètres S et w. Nous avons alors :

$$R_{x_1}(t) = w_1 \varphi_1(t) \otimes P_{x_1}(t) \tag{2.2}$$

où

$$P_{x_1}(t) = \sum_{i=0}^{\infty} \delta(t - \theta_1 - i x_1)$$

et les formules des différents noyaux $\varphi_1(t)$ sont définies dans la sous-section 2.3. Le calcul de la réponse se fait alors indépendamment pour les différentes entrées puis la somme est faite pour obtenir la réponse totale. Cette transformation permet d'utiliser l'outil de convolution discrète disponible dans les librairies de Matlab et d'optimiser grandement l'implémentation du quantron en se débarrassant des boucles itératives et des relations logiques peu efficaces de la première tentative.

2.2 Contraintes de la discrétisation

La nature informatique de l'approximation impose sa première contrainte : afin d'utiliser l'outil de convolution discrète, nous devons approximer le peigne $P_{x_1}(t)$ et le noyau $\varphi_1(t)$ par des vecteurs $\overrightarrow{P_{x_1}(t)}$ et $\overrightarrow{\varphi_1(t)}$. Il s'agit d'effectuer une forme d'échantillonnage des fonctions continues pour les transposer sur des vecteurs d'une longueur à déterminer. Nous définissons « $nbpts$ » le nombre de points, ou d'éléments, du vecteur $\overrightarrow{P_{x_1}(t)}$. Le nombre d'éléments du vecteur noyau $\overrightarrow{\varphi_1(t)}$, plus petit, est alors déterminé de façon à respecter l'échelle du temps. $nbpts$ devient ainsi un nouveau paramètre caché de l'approximation que nous fixons au départ à 1000. Cette valeur a été choisie empiriquement afin d'obtenir une réponse totale visuellement acceptable et de ne pas trop ralentir les calculs. Nous nous rendons d'ailleurs compte que le temps d'exécution est proportionnel au carré de $nbpts$ lors de l'étude de son importance dans un prochain chapitre.

Par ailleurs, afin d'effectuer la convolution des vecteurs du peigne $\overrightarrow{P_{x_1}(t)}$ et du noyau $\overrightarrow{\varphi_1(t)}$ il est impossible d'utiliser un support infini. C'est-à-dire qu'il est nécessaire de limiter le nombre d'impulsions du peigne et donc d'utiliser un peigne tronqué. Nous devons remplacer l'infini par un paramètre N à la définition de $P_{x_1}(t)$ afin d'obtenir :

$$P_{x_1}(t) = \sum_{i=0}^{N} \delta(t - \theta_1 - ix_1). \tag{2.3}$$

Ceci implique dans le modèle du quantron une quantité limitée de neurotransmetteurs disponibles et par le fait même un nombre limité d'impulsions transmissibles. Cette limitation n'est pas étrangère aux travaux précédents effectués sur ce modèle. Il n'en demeure pas moins qu'un nouveau paramètre N apparaît et doit être pris en considération. Pour l'instant, nous fixons la valeur de ce paramètre à $N = 10$ et ce pour toutes les entrées de tous les quantrons. Cette valeur est choisie suffisamment grande pour permettre des superpositions de noyaux nécessaires à une grande variété de résultats, mais suffisamment petite de sorte à ne pas ralentir excessivement la vitesse des calculs. Les effets du paramètre et de ce choix sont également présentés dans un prochain chapitre.

2.3 Approximations des noyaux

Une autre approximation suggérée par Labib (Labib, 1999) se situe au niveau de la forme des noyaux utilisés. La formule originale obtenue par l'étude de la diffusion des neurotransmetteurs produit un noyau dont l'apparence rappelle celle d'une vague. C'est d'ailleurs le nom que nous assignerons à ce type de noyau. Il est aussi suggéré de simplifier le modèle en utilisant un noyau de forme carrée, qui est utilisée ensuite pour démontrer la résolution du problème XOR à l'aide d'un seul quantron. En plus de celles-ci, nous allons définir deux autres formes intermédiaires qui sont le triangle et la parabole. Afin d'être utilisées dans la convolution, nous définissons les formes des noyaux à l'aide des formules suivantes.

Pour le noyau « Carré » :

$$\varphi_C(t) = u(t) - u(t - S). \tag{2.4}$$

Pour le noyau « Triangle » :

$$\begin{aligned}\varphi_T(t) &= [u(t) - u(t-S)]t + [u(t-S) - (t-2S)](2S-t) \\ &= t \times u(t) + [2S-2t]u(t-S) + [t-2S]u(t-2S).\end{aligned} \tag{2.5}$$

Pour le noyau « Parabole » :

$$\varphi_P(t) = [u(t) - u(t-2S)]p(t) \tag{2.6}$$

où nous voulons $p(t)$ une parabole concave ayant pour zéros $z_1 = 0$ et $z_2 = 2S$:

$$p(t) = -t(t-2S) = -t^2 + 2St.$$

Pour le noyau « Vague » :

$$\varphi_V(t) = V(t) + \big(V(S) - V(t) - V(t-S)\big)u(t-S) + \big(V(t-S) - V(S)\big)u(t-2S) \tag{2.7}$$

où

$$V(t) = 2\left(1 - \Phi\left(\frac{\ln a}{\sqrt{t}}\right)\right).$$

Selon ces formules, il s'en suit que les noyaux ont les maximums aux valeurs $t = S$ suivants :

$$\varphi_C(S) = 1$$

$$\varphi_T(S) = S$$

$$\varphi_P(S) = S^2$$

$$\varphi_V(S) = V(S) = 2\left(1 - \Phi\left(\frac{\ln a}{\sqrt{S}}\right)\right). \tag{2.8}$$

Dans trois cas, nous avons un maximum qui est fonction de la largeur S du noyau. Afin de se débarrasser de cette dépendance, nous avons normalisé les formules des noyaux afin que leurs maximums soient toujours égaux à 1. Ainsi :

$$\varphi_C^*(t) = u(t) - u(t-S) \tag{2.9}$$

$$\varphi_T^*(t) = \frac{1}{S}\{t \times u(t) + [2S-2t]u(t-S) + [t-2S]u(t-2S)\} \tag{2.10}$$

$$\varphi_P^*(t) = \frac{1}{S^2}[u(t) - u(t-2S)](-t^2 + 2St) \tag{2.11}$$

$$\varphi_V^*(t) = V^*(t) + \big(V^*(S) - V^*(t) - V^*(t-S)\big)u(t-S)$$
$$+\big(V^*(t-S) - V^*(S)\big)u(t-2S) \qquad (2.12)$$

où

$$V^*(t) = \frac{\left(1 - \Phi\left(\frac{\ln a}{\sqrt{t}}\right)\right)}{\left(1 - \Phi\left(\frac{\ln a}{\sqrt{S}}\right)\right)}.$$

Nous nous permettons cette opération considérant que les mêmes résultats peuvent êtres obtenus à la sortie des quantrons en modifiant le paramètre w du même facteur multiplicatif. Il reste dans la formule de $\varphi_V^*(t)$ une constante « a » que nous devons fixer. Dans le modèle de diffusion du quantron, cette valeur représente la distance à parcourir par les neurotransmetteurs pour atteindre les récepteurs de la membrane postsynaptique. Afin d'avoir une forme telle qu'obtenue originalement, la valeur de a doit être entre 1 et 2. Nous avons choisi la valeur 1,25 qui a été conservée pour tous les essais subséquents utilisant le noyau Vague. Notons ici que les largeurs finales des noyaux $\varphi_T^*(t), \varphi_P^*(t)$ et $\varphi_V^*(t)$ sont de $2S$, S étant le temps de leur début jusqu'à leur maximum. L'image suivante permet de visualiser la forme des différents noyaux.

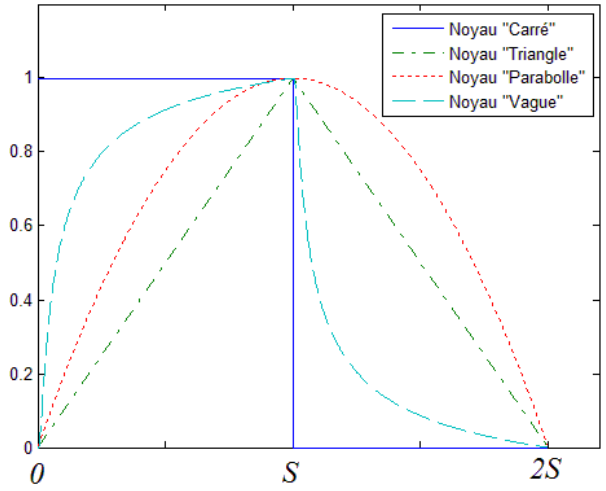

Figure 2.1 : Illustration de la forme des quatre noyaux

2.4 Entrées et sorties

L'utilisation de la convolution et des formules que nous venons de définir nous permet d'obtenir sous la forme d'un vecteur la réponse totale de la membrane postsynaptique. Selon le modèle, il y a déclenchement d'un potentiel d'action à la sortie du quantron si cette réponse dépasse le seuil d'activation Γ. Le vecteur est donc parcouru afin de détecter la présence d'un dépassement du seuil. Pour nos besoins, une première sortie binaire est activée si le seuil est dépassé au moins une fois. En prévision d'applications futures, une seconde sortie indique le temps associé au premier dépassement détecté.

À l'intérieur d'un MLQ sans l'utilisation de l'approximation binaire de la sortie, c'est le temps requis au déclenchement du potentiel d'action, soit notre deuxième sortie, qui est directement utilisé comme entrée aux couches suivantes. Puisque nous utilisons plutôt la sortie binaire, où 1 représente la présence d'un signal et 0 l'absence, il est nécessaire d'effectuer une modification à l'interprétation d'une entrée de valeur 0. Étant donné que l'entrée est ici généralement interprétée comme la période du signal, ou la distance entre les noyaux de la réponse, il est nécessaire de forcer l'interprétation de l'entrée 0 comme l'absence totale de signaux plutôt que d'une période nulle et donc d'une fréquence infinie. Ceci est codé directement dans l'algorithme.

2.5 Le problème du XOR

Afin de confirmer la concordance avec les résultats obtenus par Labib, nous avons reproduit la solution proposée au problème du XOR. Il s'agit ici de démontrer la capacité du quantron à catégoriser quatre ensembles de coordonnées non linéairement séparables. Nous considérons l'ensemble des coordonnées générales de quatre points représentant un ou exclusif quelconque :

$$\{(x,x);(y,y);(x,y);(y,x)\}. \tag{2.13}$$

Il est suggéré qu'une solution à ce problème se trouve dans l'ensemble de paramètres suivants pour un quantron à deux entrées avec le noyau « Carré ».

$$\begin{pmatrix} w_1 \\ S_1 \\ \theta_1 \\ w_2 \\ S_2 \\ \theta_2 \end{pmatrix} = \begin{pmatrix} \Gamma/2 \\ min(x,y)/2 \\ 0 \\ \Gamma/2 \\ min(x,y)/2 \\ min(x,y)/2 \end{pmatrix} \tag{2.14}$$

Nous obtenons par exemple avec les valeurs $x = 0{,}3,\ y = 0{,}7$ et $\Gamma = 1$ les paramètres :

$$\begin{pmatrix} w_1 \\ S_1 \\ \theta_1 \\ w_2 \\ S_2 \\ \theta_2 \end{pmatrix} = \begin{pmatrix} 0{,}5 \\ 0{,}15 \\ 0 \\ 0{,}5 \\ 0{,}15 \\ 0{,}15 \end{pmatrix}. \tag{2.15}$$

Avec ces paramètres, l'algorithme du quantron produit toutefois quatre sorties actives alors que nous nous attendions à une réponse $\{0; 0; 1; 1\}$ suivant l'ordre précédent des entrées. C'est en examinant les graphiques des signaux que nous observons la raison de cette erreur.

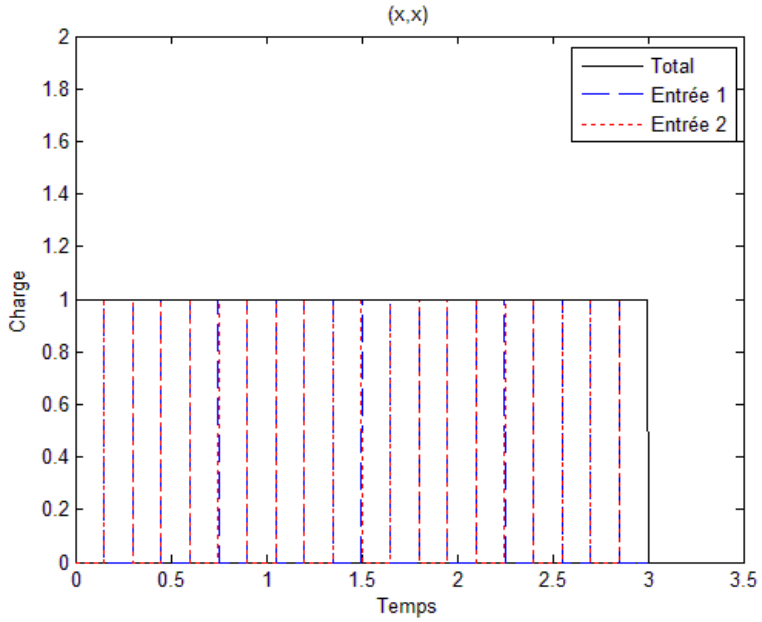

Figure 2.2 : Graphique théorique des signaux correspondant à l'entrée (x, x)du problème XOR

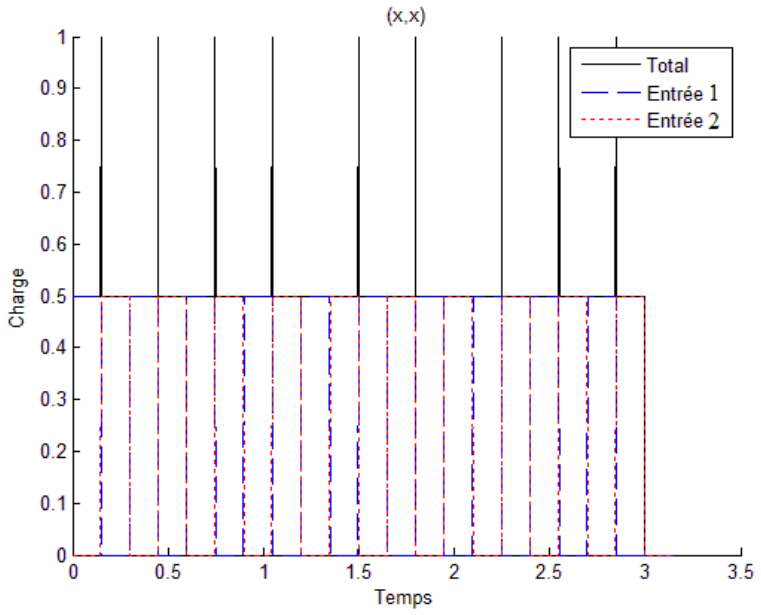

Figure 2.3 : Graphique approximé des signaux correspondant à l'entrée (x, x) du problème XOR avant la correction

Nous voyons que dans une des situations où nous attendions selon le modèle théorique à une absence d'activation, les noyaux se chevauchent parfois à leurs extrémités. C'est en fait une première observation de l'effet de la résolution de l'approximation discrète des signaux. Selon les paramètres utilisés, les noyaux carrés devraient se suivre très exactement, et ce sans espace. Comme ils sont composés de fonctions échelons, la limite à droite d'un noyau carré est égale à zéro alors que la limite gauche est égale à un. Leur somme en ce point de rencontre donne donc 1, ou 0,5 considérant les poids, et donc n'atteint pas le seuil et n'active pas la sortie. Par contre lors de l'échantillonnage nécessaire à la discrétisation des signaux, les largeurs des noyaux carrés, en nombres d'éléments du vecteur utilisé, sont appelés à varier de plus ou moins un selon le $nbpts$ utilisé. Le premier et le dernier élément possible du vecteur sur lequel se trouve le noyau sont assignés soit à 1 ou à 0 selon la fraction de l'échelle continue qu'ils représentent qui sont couvertes par le noyau carré. C'est pour cette raison que nous obtenons un court chevauchement d'une ou deux unités alors que nous ne nous attendions à aucun. Pour contourner ce problème, il est envisageable de retirer une petite valeur aux paramètres de largeur des noyaux « S », par

exemple 0,01. C'est avec les nouveaux paramètres suivants que nous obtenons la réponse attendue $\{0; 0; 1; 1\}$ et les signaux qui suivent.

$$\begin{pmatrix} w_1 \\ S_1 \\ \theta_1 \\ w_2 \\ S_2 \\ \theta_2 \end{pmatrix} = \begin{pmatrix} 0{,}5 \\ 0{,}14 \\ 0 \\ 0{,}5 \\ 0{,}14 \\ 0{,}15 \end{pmatrix} \tag{2.16}$$

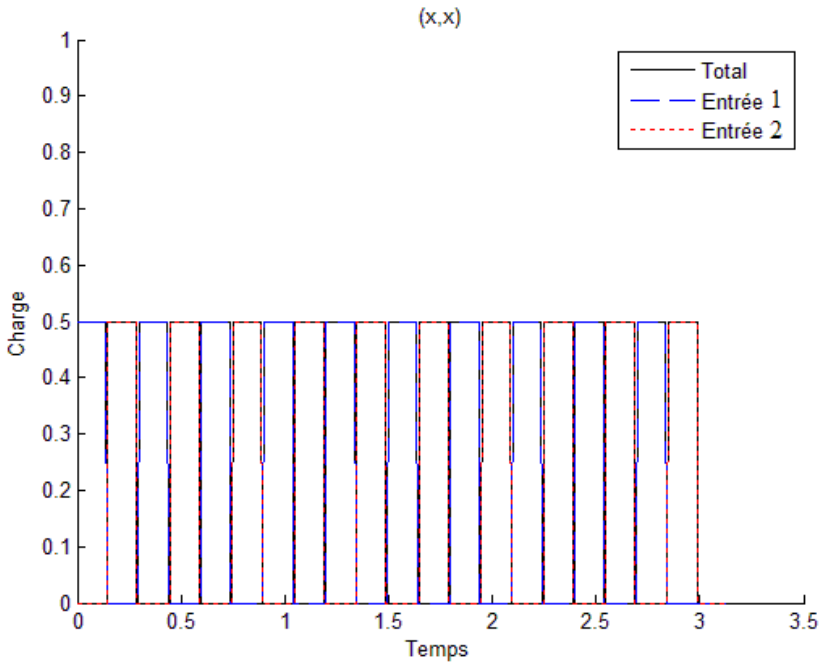

Figure 2.4 : Graphique approximé des signaux correspondant à l'entrée (x, x) du problème XOR après la correction

Nous notons ici une remarque par rapport à la solution générale du problème. Cette solution a été testée avec une grande quantité de différentes valeurs pour x et y. Il s'avère que la solution fonctionne dans la majorité des cas. Par contre, lorsque la plus grande des deux valeurs est un multiple entier de la plus petite, les deux signaux s'entrecroisent exactement et la superposition nécessaire pour obtenir l'activation aux deux points voulus n'a pas lieu. Voici un exemple de signaux pour un couple de valeurs $x = 0{,}3,\ y = 0{,}6$ et $\Gamma = 1$ et une entrée (x, y).

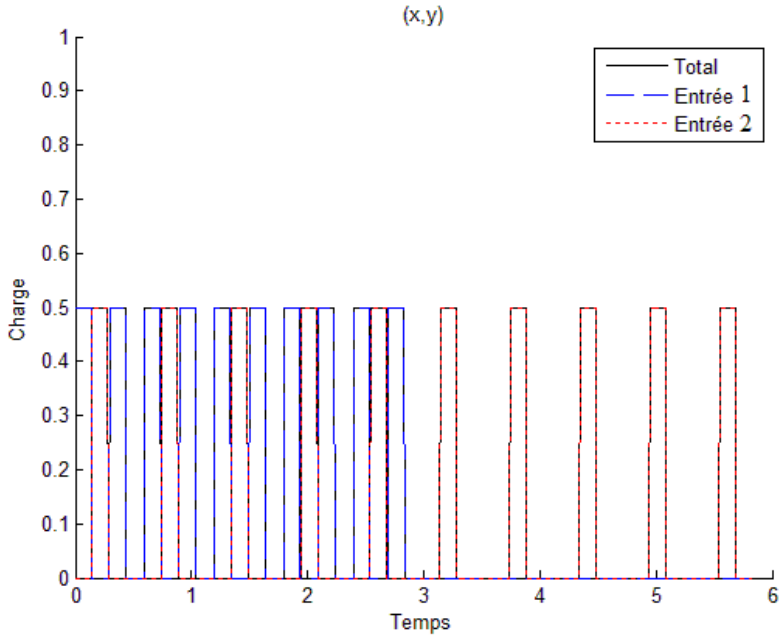

Figure 2.5 : Graphique des signaux pour un couple de valeurs où la solution n'est pas valide.

Ce premier test de l'algorithme confirme la concordance avec les résultats théoriques. Par contre, il nous avertit bien de faire attention aux cas limites de superposition, à l'effet de la résolution et du paramètre $nbpts$. Nous pouvons tout de même nous attendre à ce que les cas limites de superposition soient moins fréquents avec les autres formes de noyaux qui ne possèdent pas des frontières aussi abruptes.

3 PRODUCTION D'IMAGES ALÉATOIRES

Le modèle du quantron ayant été implémenté avec succès, il a été possible d'en étudier les caractéristiques. Afin d'obtenir une première impression de son comportement dans le cadre de la production d'images, nous avons décidé de tirer profit des avantages de l'implémentation informatique qui nous permet de produire automatiquement et rapidement une grande quantité de résultats. Nous désirions donc concevoir un algorithme capable de produire en grande quantité des images que nous pouvions étudier et combiner afin de produire les lettres de l'alphabet.

3.1 Plages de variabilité des paramètres

Afin de produire des images variées, il est entendu qu'il est nécessaire de varier les paramètres des quantrons qui les produisent. Considérant le paramètre N fixé à dix noyaux et la résolution $nbpts$ à 1000pts pour l'instant, il n'en demeure pas moins une liste importante de paramètres qui peuvent prendre des valeurs variées.

1. Noyau : la forme choisie pour le noyau, soit « Carré », « Triangle », « Parabole » et « Vague ».
2. n : le nombre d'entrées du quantron produisant l'image. Pour l'instant considéré comme $n = 2$ pour les deux coordonnées (x, y) des pixels de l'image, nous verrons plus tard que contrairement au perceptron, la même coordonnée peut être utilisée plus d'une fois comme entrée et changer le résultat.
3. $\overrightarrow{W}$: le vecteur de poids des entrées. Nous avons ici $\overrightarrow{W} \in \mathbb{R}^n$, chacun des éléments du vecteur de poids peut en théorie prendre n'importe quelle valeur réelle.
4. $\vec{S}$: le vecteur des temps d'interruption, ou largeurs de noyaux. Nous avons $\vec{S} \in \mathbb{R}^{+n}$, chacun des éléments peut prendre n'importe quelle valeur réelle positive.
5. $\overrightarrow{\Theta}$: le vecteur des délais. Nous avons $\overrightarrow{\Theta} \in \mathbb{R}^{+n}$, chacun des éléments peut prendre n'importe quelle valeur réelle positive.
6. Γ : le seuil nécessaire au déclenchement d'un potentiel d'action. Nous avons $\Gamma \in \mathbb{R}$, le seuil peut prendre n'importe quelle valeur réelle.
7. (X_{max}, Y_{max}) : les valeurs maximales des coordonnées aux bords de l'image. Nous avons $(X_{max}, Y_{max}) \in \mathbb{R}^{+2}$, les valeurs maximales doivent être réelles et positives.

8. (L_x, L_y) : le nombre de pixels dans l'image. Nous avons $(L_x, L_y) \in \mathbb{N}^{+2}$, les nombres de pixels dans les axes X et Y doivent êtres entiers et positifs.

Nous avons une très grande variété de paramètres possibles à explorer. Plusieurs de ceux-ci font partie d'un ensemble non seulement infini, mais également continu. Pour ce qui est de l'aspect continu des ensembles, nous devions déterminer une méthode de sélection de valeurs, soit par intervalles équidistants ou par échantillonnage aléatoire, mais nous devions d'abord éliminer l'aspect infini des ensembles.

Il est facile d'imaginer que ce n'est pas la totalité des combinaisons possibles des paramètres qui produisent des images intéressantes. Un exemple évident serait l'utilisation de poids $\overrightarrow{W}$ très élevés par rapport au seuil Γ. Il n'y aurait alors aucun besoin de superposition de signaux pour atteindre le seuil et il y aurait activation de la sortie, peu importe les valeurs des entrées. Ceci produirait une image entièrement blanche et dépourvue d'intérêt. Nous sommes donc menés à tenter de limiter les combinaisons de paramètres qui produisent des images inutiles afin de maximiser le temps de recherche. Pour ce faire, nous utilisons nos connaissances préalables du fonctionnement du quantron afin d'émettre des lemmes sur les relations entre les paramètres. Ces lemmes nous permettent alors de déterminer des règles qui limitent les plages de valeurs à explorer.

3.2 Lemmes

Lemme 1 : il est possible d'obtenir exactement les mêmes résultats avec deux seuils Γ différents si l'on modifie le vecteur de poids $\overrightarrow{W}$ par le même facteur multiplicatif qui différencie les seuils.

Preuve : Si l'on double la valeur du seuil à atteindre par les signaux, il suffit de doubler les poids des entrées, et par le fait même les amplitudes des signaux, pour arriver à la même réponse. ■

Ce lemme permet de fixer la valeur du seuil à une constante et de transférer sa variabilité aux valeurs des poids. Nous choisissons ici par souci de simplicité la valeur $\Gamma = 1$ que nous conservons dorénavant pour tous les essais.

Lemme 2 : Les mêmes résultats avec un couple d'entrées peuvent être obtenus avec un autre couple dans la même proportion si l'on multiplie les paramètres $\vec{S}$ et $\vec{\Theta}$ par le même facteur qui différencie les deux couples.

Preuve : Il est possible d'obtenir la même sortie avec les entrées (x_1, y_1) et $(x_2, y_2) = k(x_1, y_1)$ si l'on modifie les paramètres de largeur et de délais de telle sorte que $\overrightarrow{S_2} = k\overrightarrow{S_1}$ et $\overrightarrow{\Theta_2} = k\overrightarrow{\Theta_1}$. Ceci est par contre vrai uniquement pour la sortie binaire. Dans le cas de la sortie qui représente le temps requis pour l'activation, celle-ci est aussi multipliée par le facteur k. Cette relation correspond en fait à un simple changement de l'échelle du temps sur laquelle se déroule le déploiement des signaux. ■

En imposant d'abord des bornes supérieures (X_{max}, Y_{max}) toutes deux à un, nous avons distribué linéairement les coordonnées en pixels de l'image selon leurs positions dans une matrice de $L_x \times L_y$ éléments de la façon suivante.

$$\begin{bmatrix} \left(\frac{1}{L_x}, \frac{1}{L_y}\right) & \left(\frac{1}{L_x}, \frac{2}{L_y}\right) & \cdots & \left(\frac{1}{L_x}, 1\right) \\ \left(\frac{2}{L_x}, \frac{1}{L_y}\right) & \left(\frac{2}{L_x}, \frac{2}{L_y}\right) & & \vdots \\ \vdots & & \ddots & \\ \left(1, \frac{1}{L_y}\right) & \cdots & & (1,1) \end{bmatrix} \tag{3.1}$$

Cette limitation semble à première vue arbitraire. Par contre, si l'on considère le deuxième lemme, nous comprenons qu'il serait possible d'obtenir les mêmes images avec une limite supérieure de par exemple 2 plutôt que 1, si l'on doublait tous les paramètres $\vec{S}\ et\ \vec{\Theta}$. Or, encore une fois par souci de simplicité, nous avons décidé d'utiliser la valeur unitaire.

La configuration des valeurs d'entrées en fonction des coordonnées des pixels telle que proposée ci-dessus permet aussi de choisir différentes valeurs de (L_x, L_y) tout en conservant la même image. Il s'agit alors simplement de choisir la résolution de l'image en nombre de pixels. Nous avons choisi pour des fins de simplicité de créer des images carrées et donc $L_x = L_y = L$. Un choix différent produirait une image similaire, mais rectangulaire, avec un des axes compressé.

Ensuite, le choix de la résolution de l'image peut en théorie être aussi grand que désiré. Il est par contre important de prendre en considération le temps que prend la production des images que nous voulions en grand nombre. Comme chaque pixel requiert le calcul de la sortie d'un quantron, ou de plusieurs lorsque nous utilisons des réseaux, le temps d'exécution est proportionnel au carré de la résolution. Aussi, selon la définition des valeurs d'entrées, plus la valeur de L est grande, plus les premières valeurs, par exemple $\left(\frac{1}{L_x}, \frac{1}{L_y}\right)$, seront petites. Des entrées, ou périodes entre les noyaux des signaux, trop petites par rapport à 1 risquent de forcer la superposition des noyaux indépendamment des autres paramètres et de créer des bandes d'activation du quantron là où elles ne seront pas nécessairement désirées. Plusieurs valeurs de résolution d'images L ont été utilisées lors des expérimentations de ce travail. C'est par expérience et en considérant les deux observations dévoilées ci-dessus que nous avons fixé la valeur $L = 20$ comme standard pour les images finales du travail. Cette valeur permet suffisamment de détails aux images tout en permettant une production rapide et en limitant les bandes indésirables. Il est à noté que l'utilisation des coordonnées matricielles fait en sorte que les valeurs de l'entrée X augmentent du haut vers le bas de l'image et celles de l'entrée Y de la gauche vers la droite.

Lemme 3 : Les mêmes résultats peuvent être obtenus avec un vecteur de délais $\overrightarrow{\Theta_1}$ et un autre vecteur de délais $\overrightarrow{\Theta_2}$ auquel nous avons ajouté la même constante à chacun de ses éléments.

Preuve : Comme ce vecteur contrôle les délais dans le temps avant les débuts des premiers noyaux de chaque entrée, rajouter un délai supplémentaire égal pour les deux entrées n'affecte pas la superposition des signaux et la présence ou non d'un dépassement du seuil d'activation. La sortie binaire se trouve inchangée, mais la sortie temporelle se trouve décalée par le même délai.■

Il s'avère ainsi inutile d'utiliser des valeurs de délais qui sont toutes non nulles. Par exemple dans le cas de deux entrées, les deux vecteurs de délais $\overrightarrow{\Theta_1} = (\theta_x, \theta_y)$ et $\overrightarrow{\Theta_2} = (\theta_x - \theta_x, \theta_y - \theta_x) = (0, \theta_y - \theta_x)$ sont équivalents. Il est donc possible de garder l'une des valeurs du vecteur de délais

à zéro et de ne faire varier que les autres. Pour les images créées avec des quantrons à deux entrées, nous avons choisi de fixer $\theta_x = 0$. Il est alors pris en considération que toutes les images produites peuvent être inversées par rapport à la diagonale $x = y$ de l'image en inversant les valeurs des éléments des vecteurs $\overrightarrow{W}, \vec{S}\ et\ \overrightarrow{\Theta}$.

Nous pouvons aussi imposer une limite supérieure à l'élément non nul. Puisque la valeur maximale de la première entrée est $x = 1$ et que le nombre de noyaux est limité à $N = 10$, nous savons que le dernier noyau du signal x débute au temps $t = 9$ et se termine à $t = 9 + 2S_x$. Il est ainsi inutile de tester des valeurs de θ_y plus grandes que $N - 1 + 2s_x$ car il ne se produit alors aucune superposition des deux signaux dans toute l'image. Un tel manque d'interaction entre les entrées ne peut que produire des bandes verticales et horizontales.

Lemme 4 : Si les éléments du vecteur de poids $\overrightarrow{W}$ sont strictement positifs, il est inutile de tester des valeurs supérieures à $\Gamma = 1$.

Preuve : Comme nous avons normalisé les hauteurs des noyaux, les maximums de ceux-ci se trouvent à être exactement égaux au paramètre de poids w qui les régit. Ainsi, un poids supérieur au seuil activerait la sortie, peu importe les entrées, et produirait une image entièrement blanche et sans intérêt. ■

Similairement, il est inutile de tester des vecteurs de poids $\overrightarrow{W}$ où tous les éléments sont négatifs. Dans un tel cas, avec le seuil fixé à 1, il ne se produirait jamais d'activation et nous obtiendrions une image entièrement noire et tout aussi dépourvue d'intérêt. Nous pourrions considérer un seuil Γ négatif, mais nous nous retrouverions avec une situation équivalente à un seuil positif avec des poids strictement positifs.

Il reste à considérer les vecteurs de poids que nous appelons mixtes, qui ont une valeur positive et l'autre négative. Nous pouvons ici utiliser des valeurs supérieures au seuil, car le signal au poids négatif a un effet d'atténuation sur le signal positif. Par contre, en considérant $\theta_x = 0$ tel que mentionné en conséquence du lemme 3, nous pouvons imaginer qu'une valeur de $w_x > 1$ risque fort de produire une activation totale. C'est en fait le cas pour toute valeur de $\theta_y > S_x$ où le

second signal ne débute pas assez rapidement pour atténuer le sommet du premier noyau positif. Nous pouvons aussi imaginer qu'une valeur négative d'un des éléments trop grande par rapport à la valeur positive de l'autre risque de l'atténuer trop fortement et de ne pas créer des interactions pertinentes. En fonction de ces considérations, nous avons établi les règles suivantes pour les valeurs du vecteur de poids $\overrightarrow{W}$.

a. En premier lieu, la valeur de w_x est déterminée dans l'intervalle $[-1,1]$.
b. Ensuite, a valeur de w_y est déterminée en fonction de w_x dans l'intervalle $[-1,1]$ si $w_x \geq 0$ et dans l'intervalle $[0,2]$ si $w_x < 0$.

Ces règles respectent le lemme 4 et les considérations sur les vecteurs mixtes. Nous sommes conscients que ces considérations ne sont pas absolues et que les règles qui en découlent risquent d'éliminer certaines combinaisons qui auraient tout de même produit des images non triviales. Par contre, l'alternative pour n'en éliminer aucune serait de permettre $w_x \in \mathbb{R}$ et ensuite $w_y \in \mathbb{R}$ ou $w_y \in \mathbb{R}^+$ selon w_x. Cette alternative ne règle pas le problème des ensembles infinis et produirait une grande majorité d'images triviales noires ou blanches. C'est donc la première alternative qui a été conservée.

Une règle similaire à la précédente a été imposée aux valeurs du vecteur des largeurs de noyaux. Il est considéré que cette largeur ne peut être inférieure à zéro. Une valeur négative pourrait correspondre pour les signaux à une inversion des formes par rapport à leur point de départ. Ceci reviendrait à un simple décalage négatif du délais θ de $-2S$ dans le cas des noyaux « Carré », « Triangle » et « Parabole » qui sont symétriques. Un résultat différent pourrait être obtenu avec le noyau « Vague » mais la valeur de S est, selon le modèle théorique, un temps d'interruption qui, négatif, n'a pas de signification logique. Nous avons donc une première limite inférieure pour les valeurs de ce paramètre.

Nous savons selon la convention adoptée pour les coordonnées de l'image que la plus petite valeur d'entrée est de $\frac{1}{L} = 0{,}05$ pour $L = 20$. Nous pouvons imaginer qu'une valeur de S inférieure 0,025 impliquerait qu'il n'y ai aucune contribution des noyaux avec ses voisins du même signal, et ce, pour toute l'image. Cette situation risque fort de ne pas produire d'images

intéressantes et impose une nouvelle limite inférieure aux valeurs de S. En pratique, cette nouvelle limite est suffisamment proche de zéro pour être ignorée.

Nous savons aussi, selon la même convention, que la valeur maximale d'entrée est de 1. Dans ce cas, avec $2S = N = 10$, ou $S = N = 10$ pour le noyau « Carré », nous sommes certains que la fin du premier noyau d'un signal atteindra le début du dernier, et ce, pour toute l'image. Avec cette limite supérieure, nous sommes certains d'inclure la valeur de S qui maximise le sommet qu'un signal peut atteindre par recouvrement avec lui-même. À noter qu'en pratique, nous avons observé que cette limite supérieure est extrêmement conservatrice et produit trop d'images sans intérêt. Elle a éventuellement été abaissée à $2S = N/2 = 5$.

Les nouvelles limites sont les suivantes.

1. Noyau : la forme choisie pour le noyau, soit « Carré », « Triangle », « Parabole » et « Vague ». Ces quatre formes ont été utilisées et les différentes images produites comparées.
2. n : le nombre d'entrées du quantron produisant l'image. Au départ considéré comme $n = 2$ pour les deux coordonnées (x, y) des pixels de l'image, nous avons plus tard vu que contrairement au perceptron, la même coordonnée peut être utilisée plus d'une fois comme entrée et changer le résultat.
3. $\overrightarrow{W}$: le vecteur de poids des entrées. Nous avons :
    a. En premier lieu, la valeur de w_x est déterminée dans l'intervalle $[-1,1]$.
    b. Ensuite, a valeur de w_y est déterminée en fonction de w_x dans l'intervalle $[-1,1]$ si $w_x \geq 0$ et dans l'intervalle $[0,2]$ si $w_x < 0$.
4. $\vec{S}$: le vecteur des temps d'interruption, ou largeurs de noyaux. Nous avons $S_{x,y} \in [0, 5]$, ensuite modifié à $S_{x,y} \in [0, 2{,}5]$.
5. $\overrightarrow{\Theta}$: le vecteur des délais. Nous avons $\theta_x = 0 \; et \; \theta_y < 9 + 2S_x$.
6. Γ : le seuil nécessaire au déclenchement d'un potentiel d'action. Nous avons $\Gamma = 1$.
7. (X_{max}, Y_{max}) : les valeurs maximales des coordonnées aux bords de l'image. Nous avons $(X_{max}, Y_{max}) = (1,1)$.
8. (L_x, L_y) : le nombre de pixels dans l'image. Nous avons $(L_x, L_y) = (20, 20)$.

Nous résumons les changements de plages de paramètres dans le tableau suivant.

Tableau 3-1 : Plages de paramètres initiales et modifiées à la suite des règles établies

Paramètre	*Plage initiale*	*Plage modifiée*
Noyau	Carré, Triangle, Parabole, Vague	Carré, Triangle, Parabole, Vague
n	2	2
$\overrightarrow{W}$	$\mathbb{R}^n$	$[-1,1]$ ou $[0,2]$
$\vec{S}$	$\mathbb{R}^{+n}$	$[0,2{,}5]$
$\overrightarrow{\Theta}$	$\mathbb{R}^{+n}$	$[0, 9+2s_x]$
Γ	$\mathbb{R}$	1
(X_{max}, Y_{max})	$\mathbb{R}^{+2}$	$(1,1)$
(L_x, L_y)	$\mathbb{N}^{+2}$	$(20,20)$

3.3 Méthode aléatoire

Les nouvelles plages de valeurs ne sont plus infinies, mais certaines d'entre elles demeurent continues. En considérant le nombre d'entrées $n = 2$, nous avons un ensemble de cinq paramètres continus $\{w_x, w_y, s_x, s_y, \theta_y\}$ à faire varier. Une possibilité aurait été de choisir pour chaque paramètre une quantité Q de valeurs uniformément réparties dans leurs plages respectives pour ensuite tester toutes les combinaisons possibles. Une telle méthode aurait produite Q^5 images, ou même $4Q^5$ si l'on considère les quatre noyaux possibles. Nous nous serions rapidement retrouvés avec une très grande quantité d'images à produire avant d'avoir une idée des images possibles. Par exemple, même avec $Q = 4$ nous aurions $4Q^5 = 4096$. Si nous avions voulu créer d'autres images par la même méthode, il aurait fallu augmenter le nombre d'échantillons par exemple à 5 ou 6 pour obtenir respectivement 12500 et 31104 images. Cette méthode de production n'est pas adaptée à nos besoins et nous avons cherché une alternative.

Plutôt que de désirer produire à chaque fois un échantillon global des images possibles, nous avons opté pour un échantillonnage aléatoire. Chacune des plages de valeurs a été traitée comme le domaine d'une loi de probabilité uniforme. Pour chaque nouvelle image désirée, un algorithme pseudoaléatoire détermine une valeur pour chaque paramètre variable et les transmet à l'approximation du quantron qui produit l'image correspondante. Cette méthode permet de

produire à chaque fois la quantité d'images que l'on désire sans se soucier de la répétition des images déjà produites. L'outil pseudoaléatoire Matlab est une version modifiée de l'algorithme « Soustraction avec emprunt de Marsaglia ». La méthode génère toutes les valeurs de précision double dans l'intervalle $[2^{-53}, 1-2^{-53}]$, et peut produire en théorie 2^{1492} valeurs avant de se répéter. Cet intervalle se rapproche suffisamment d'une loi uniforme $U(0,1)$ pour être traité comme tel pour nos fins. Il s'agit ensuite, pour obtenir une valeur dans un intervalle quelconque, d'effectuer l'opération :

$$U(a,b) = a + (b-a) \cdot U(0,1). \tag{3.2}$$

Nous avons donc créé un algorithme qui permet de produire sur demande une quantité variable d'images avec un quantron de deux entrées avec des paramètres $\{w_x, w_y, s_x, s_y, \theta_y\}$ pseudoaléatoires dans les intervalles spécifiés plus haut et avec une forme de noyau donnée. Nous nous sommes rapidement rendu compte que malgré les efforts pour éliminer les combinaisons de paramètres produisant des images sans intérêt, soit entièrement blanches, entièrement noires ou sans interaction utile entre les deux entrées, celles-ci demeuraient très fréquentes. Nous avons donc développé une méthode de rejet automatique de ces images afin qu'elles n'apparaissent pas dans nos banques d'images à étudier.

1. Si $\sum_{i=1}^{L}\sum_{j=2}^{L}\left|p_{i,j} - p_{i,j-1}\right| = 0$, alors l'image ne varie pas selon l'axe Y et est rejetée.
2. Si $\sum_{j=1}^{L}\sum_{i=2}^{L}\left|p_{i,j} - p_{i-1,j}\right| = 0$, alors l'image ne varie pas selon l'axe X et est rejetée.

$p_{i,j}$ étant la valeur du pixel de coordonnées i, j dans la matrice image. Lorsqu'une image est rejetée, l'algorithme choisit simplement un nouvel ensemble de paramètres et poursuit sa production. Nous avons éliminé les images blanches et noires pour des raisons évidentes. Les images qui ne varient pas selon un des axes peuvent être utiles à la production de lettres, mais celles-ci peuvent s'obtenir facilement avec un quantron d'une seule entrée. Il était donc inutile de conserver celles produites avec deux entrées.

Le travail préalable de cette section a été nécessaire à la production efficace des images que nous désirions étudier. Il a également permis une meilleure compréhension des relations entre les paramètres avant même de les observer en action. Les lemmes émis et les règles que nous en

avons tirées peuvent être utilisés dans le futur afin d'améliorer l'efficacité d'un algorithme d'apprentissage en éliminant automatiquement les changements de paramètres inutiles. La prochaine section discute des premières images obtenues.

4 PREMIÈRES IMAGES

Une première observation des résultats montre que certaines formes semblables reviennent fréquemment malgré l'uniformité des probabilités des différents paramètres. Ceci suggère une certaine stabilité de ces formes par rapport à la variation des paramètres qui en sont responsables. Nous voulons dire ici que ces formes ne sont pas fortement affectées lorsqu'on augmente ou diminue un des paramètres d'une même quantité qui fait autrement modifier grandement une forme plus rare dans nos échantillons. Cette hypothèse a été confirmée lorsque nous avons tenté d'améliorer les caractères de l'alphabet en modifiant les paramètres.

4.1 Choix du noyau

Nous observons une très grande similitude entre les résultats obtenus pour les noyaux « Triangle », « Parabole » et « Vague ». Nous retrouvons dans les trois cas les mêmes formes fréquentes et il est difficile de distinguer visuellement les résultats des trois noyaux. Par contre, les images produites par le noyau « Carré » se distinguent par leur manque de courbes et la présence plus fréquente de pixels isolés. Nous attribuons cette différence au problème de début et fin brusque du noyau discuté lors de la résolution du XOR. Les noyaux « Triangle », « Parabole » et « Carré » sont des approximations du noyau « Vague ». Comme nous ne voulons pas que la forme des noyaux soit un paramètre variable à l'intérieur même des MLQs qui produisent les images de caractères, nous avons voulu à ce stade sélectionner l'une d'elles avec laquelle nous avons travaillé exclusivement par la suite. Ceci élimine par le fait même la nécessité de produire à quatre reprises les résultats. Le noyau « Carré » est le plus simple, mais le plus différent du noyau « Vague » dans les résultats comme dans sa géométrie. Celui-ci a été le premier éliminé. Le noyau « Vague », bien qu'il soit le plus fidèle aux équations du quantron, est le plus coûteux à réaliser et les images produites prennent de deux à trois fois plus de temps de calcul comparativement aux deux autres. Comme les images qu'il produit sont très similaires aux deux autres approximations, nous nous sommes contentés de l'une d'elles pour le reste du travail. Nous estimons que la forme triangulaire est plus ressemblante au noyau « Vague » originale.

Contrairement à ces deux-ci, la courbe de dépolarisation du noyau « Parabole » n'est pas l'inverse exact de la courbe de polarisation qui précède le temps d'interruption. C'est donc sous ces considérations que nous avons conservé le noyau « Triangle » pour le reste du travail. Le tableau suivant résume les avantages et désavantages des noyaux.

Tableau 4-1 : Avantages et désavantages des formes de noyaux

Noyau	Avantages	Désavantages
Carré	Il est très simple à calculer.	C'est la forme la moins semblable au modèle original.
		Les images qu'il produit se distinguent des trois autres.
		Les débuts et fins abruptes des noyaux causent des défauts dans les images.
Triangle	Il est relativement simple à calculer.	
	Les images qu'il produit sont semblables à celles de la vague.	
	La forme de sa dépolarisation est l'inverse de celle de sa polarisation.	
Parabole	Il est plus simple à calculer que la Vague.	La forme de sa dépolarisation n'est pas l'inverse de celle de sa polarisation.
	Les images qu'il produit sont semblables à celles de la vague.	
Vague	C'est la forme originale du modèle	Le temps de calcul requis est significativement augmenté.

4.2 Premières lettres

Une fois la forme « Triangle » adoptée pour le noyau, nous avons procédé à la production d'une banque de quelques milliers d'images. Nous avons ensuite parcouru visuellement cette banque afin de sélectionner les images qui formaient des lettres entières ou des parties de lettres qui peuvent s'assembler à l'aide de couches subséquentes d'un MLQ. Contrairement aux attentes, nous observons une lacune au niveau des formes courbes qui ne sont pas très variées. Il y a aussi absence de diagonales pour une des deux orientations possibles. Ces lacunes sont explorées et

présentées au chapitre 11. Nous présentons ici les premières images de lettres obtenues entières ou assemblées.

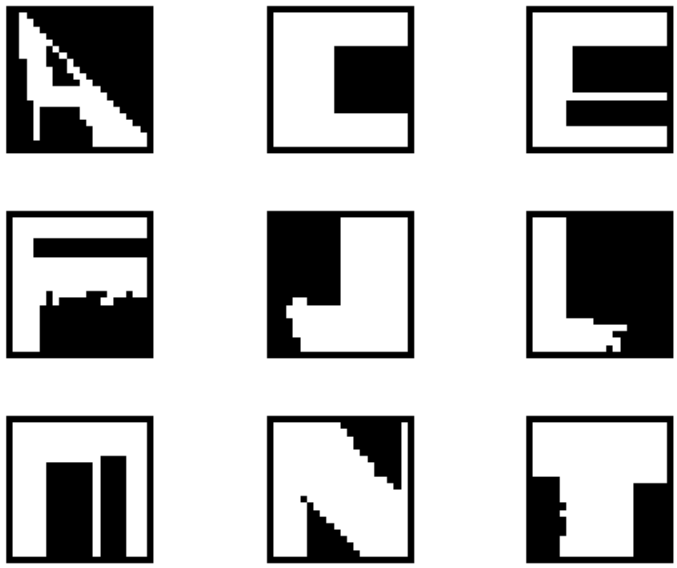

Figure 4.1 : Premières images de lettres obtenues entières ou par assemblages d'images créées par des quantrons à deux entrées

À titre d'exemple nous illustrons ici l'architecture et les paramètres du MLQ produisant la lettre A.

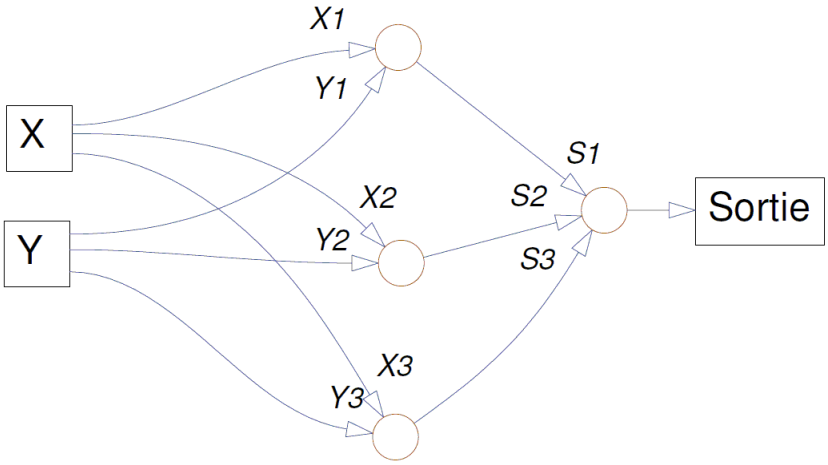

Figure 4.2 : Architecture du MLQ produisant la lettre A

Ses paramètres sont :

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} -0{,}7548 \\ 1{,}0958 \\ 1{,}6341 \\ 0{,}1266 \\ 0 \\ 1{,}7071 \end{pmatrix}, \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 0{,}4903 \\ 0{,}1416 \\ 1{,}4500 \\ 1{,}7707 \\ 5{,}4834 \\ 0 \end{pmatrix} \text{ et } \begin{pmatrix} w_{x3} \\ w_{y3} \\ S_{x3} \\ S_{y3} \\ \theta_{x3} \\ \theta_{y3} \end{pmatrix} = \begin{pmatrix} -0.3958 \\ 1{,}9577 \\ 1{,}8614 \\ 0{,}3033 \\ 0 \\ 1{,}7022 \end{pmatrix}. \quad (4.1)$$

Nous ne donnons pas ici les paramètres du quantron de sortie parce qu'à ce stade les relations logiques entre les images produites par les quantrons d'entrées sont effectués simplement dans le code. La méthode pour obtenir ces paramètres est présentée plus loin dans la section 10 et les paramètres utilisés pour toutes les lettres de la figure 4.1 sont disponibles en annexe.

4.3 Variations aléatoires

Nous voyons que la qualité des lettres n'est pas exceptionnelle. Une première tentative d'amélioration a été entamée. Le but est d'améliorer l'aspect des lettres en améliorant les images qui les composent individuellement par la modification de leurs paramètres. À ce stade, nous n'avions pas encore une bonne idée des effets des paramètres sur les images produites et l'intention était de les modifier au hasard autour de leurs valeurs originales en espérant obtenir de meilleurs résultats. Nous avons créé un algorithme auquel nous avons fourni les paramètres du MLQ associé à une image que l'on désirait améliorer et qui sélectionnait au hasard de nouveaux paramètres selon des lois de probabilité normales centrées sur les paramètres originaux. Les variances des lois normales devaient également être spécifiées. Ces variances ont été déterminées empiriquement selon les résultats obtenus.

Notons qu'à ce stade nous avions créé par curiosité un algorithme qui permet de faire la transition entre deux images produites par des quantrons. Cette transition est le résultat du passage graduel des paramètres qui ont produit la première image « A » vers ceux qui ont produit la seconde « B ».

$$\begin{bmatrix} \overrightarrow{W_\iota} \\ \overrightarrow{S_\iota} \\ \overrightarrow{\Theta_\iota} \end{bmatrix} = \begin{bmatrix} \overrightarrow{W_A} + K\left(\overrightarrow{W_B} - \overrightarrow{W_A}\right)/i \\ \overrightarrow{S_A} + K\left(\overrightarrow{S_B} - \overrightarrow{S_A}\right)/i \\ \overrightarrow{\Theta_A} + K\left(\overrightarrow{\Theta_B} - \overrightarrow{\Theta_A}\right)/i \end{bmatrix} \tag{4.2}$$

Les formules sont définies pour une transition en K images. Cette transition peut être visualisée image par image et aussi enregistrée sous la forme d'une vidéo.

Le procédé aléatoire pour l'amélioration des lettres, bien qu'instructif, ne s'est pas avéré très efficace dans son but premier. Nous avons tout de même pu observer de légères améliorations sur les caractères. Ces améliorations sont choisies selon des critères esthétiques tels la symétrie et l'égalité d'épaisseur des bandes. Voici tout de même les améliorations obtenues.

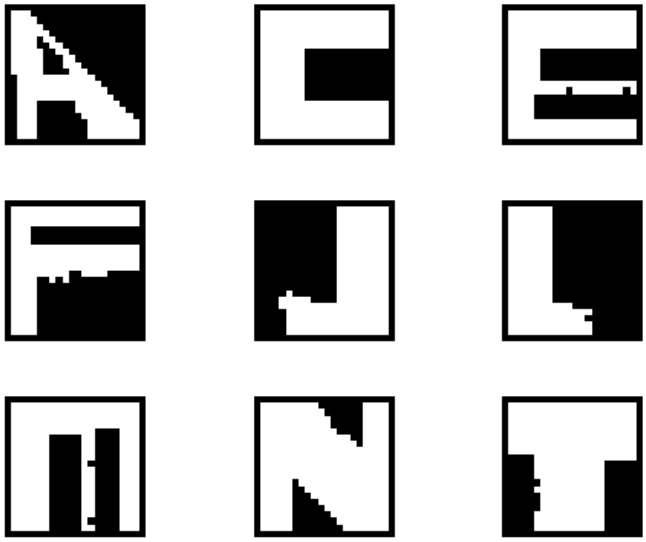

Figure 4.3 : Résultats des améliorations aléatoires sur les premières lettres produites

Les premières images obtenues nous ont d'abord permis de faire le choix d'un noyau unique. Avec ce noyau, une très grande quantité d'images ont été produites. Quelques premières lettres sont produites, mais leur qualité laisse à désirer. Une première tentative d'amélioration par changements aléatoires des paramètres n'est pas très efficace. Nous voulons donc analyser plus en profondeur les signaux qui produisent ces formes, mais d'abord nous démontrons une première propriété originale du quantron nouvellement découverte.

5 DÉDOUBLEMENT DES ENTRÉES

5.1 Traitement non linéaire

Nous discutons dans cette section d'une caractéristique du quantron que nous avons découverte et qui lui confère un avantage supplémentaire par rapport au perceptron. Contrairement à ce dernier, le quantron est capable d'utiliser plusieurs fois la même valeur d'entrée sans que le résultat soit équivalent à une utilisation unique. Pour un perceptron et une entrée unique, nous avons à la sortie y :

$$y = h(wx) \tag{5.1}$$

où $h(z)$ est la fonction d'activation, w est le paramètre de poids et x l'entrée. Nous pouvons en théorie utiliser l'entrée x une deuxième fois.

$$y = h(w_1 x + w_2 x) \tag{5.2}$$

Cependant, nous voyons rapidement que nous obtenons le même résultat avec une seule entrée où $w = w_1 + w_2$, ce qui rend la double utilisation de la valeur x redondante et inutile. Un résultat similaire est obtenu pour le quantron si l'on ne fait que modifier la valeur du poids w_x pour la nouvelle entrée. Par contre, l'ajout des paramètres de largeur du noyau s_x et du délais θ_x rendent le nouveau signal indépendant du précédent et la somme des deux signaux n'est plus reproductible avec une seule entrée.

Nous observons que cette caractéristique confère immédiatement l'avantage suivant : il est désormais possible de produire l'opération logique OU entre deux images, produites par deux quantrons, à l'intérieur même d'un seul quantron. En ajoutant au premier deux nouvelles entrées $(x_2, y_2) = (x_1, y_1)$ auxquelles l'on assigne les paramètres du second quantron, il s'agit alors d'ajouter une valeur suffisamment grande aux délais $\overrightarrow{\Theta_2} = \left(\theta_{x_2}, \theta_{y_2}\right)$ de sorte que les premiers signaux ne se superposent jamais aux nouveaux. Nous obtenons alors le même résultat avec un quantron de quatre entrées qu'avec un MLQ de trois quantrons de deux entrées. Ceci représente

une économie en structure, mais aussi du tiers sur le nombre de paramètres requis. L'image suivante représente les deux MLQs équivalents.

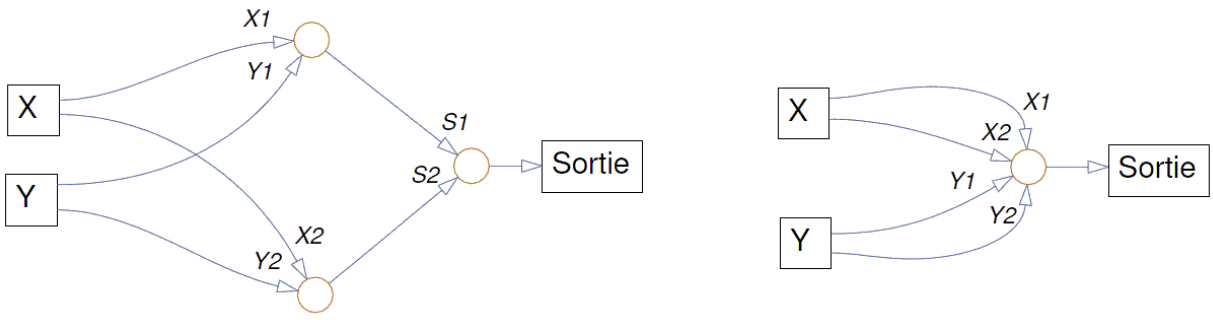

Figure 5.1 : MLQs équivalent pour l'opération OU entre deux images

Ce rapport est conservé pour chaque opération OU supplémentaire.

5.2 Intégration de la propriété

Afin de tirer profit de cette caractéristique, nous avons légèrement modifié l'algorithme de production d'images aléatoires pour pouvoir produire des images avec des entrées multiples. Les premières images que nous avons désiré obtenir devaient simplement être créées à partir d'un quantron à deux entrées qui utilisaient la même valeur. Les conditions sur les paramètres demeurent les mêmes que pour les productions précédentes. Comme une seule dimension de l'image est représentée dans les entrées du quantron, seulement un vecteur est produit. La matrice des pixels de l'image est ensuite extrapolée en répétant le vecteur L fois, permettant ainsi de produire les images plus rapidement. Les critères de rejet se résument en le suivant.

1. Si $\sum_{i=2}^{L}|p_i - p_{i-1}| \leq 1$, alors l'image n'a pas plus d'une frontière et est rejetée.

Ceci est utile pour éliminer les images qui pourraient être produites par une entrée non doublée. L'étape suivante est d'utiliser les deux entrées X et Y, puis de les doubler ou même tripler individuellement ou simultanément. Nous pouvons ainsi créer des images à partir d'un quantron « XXY », qui utilise deux fois l'entrée X et une fois Y. Nous pouvons aussi utiliser des quantrons « XYY », « $XXYY$ », « $XXXY$ », « $XXXYY$ » et ainsi de suite. Notons que, par exemple, le type

« XXY » est aussi équivalant à « XYY » pour une matrice image et des vecteurs de paramètres transposés.

Ayant plus de deux entrées, nous avons du définir de nouvelles plages de valeurs à tester pour celles-ci. Voici les règles modifiées.

1. Noyau : Triangle pour tous les signaux.
2. n : Le nombre d'entrées devient variable. $n = 3$ pour les premiers quantrons de type « XXY ».
3. $\overrightarrow{W}$: Tous les éléments du vecteur de poids peuvent prendre les valeurs dans l'intervalle $[-1,2]$. Le dernier élément est forcé positif si tous les précédents sont négatifs.
4. $\vec{S}$: Tous les éléments du vecteur des temps d'interruption prennent des valeurs dans l'intervalle $[0,\ 2{,}5]$.
5. $\overrightarrow{\Theta}$: Nous avons le premier élément du vecteur de délais $\theta_1 = 0\ et\ \theta_z < 9 + 2s_1$ pour tous les éléments subséquents.
6. Γ : Le seuil demeure fixé $\Gamma = 1$.
7. (X_{max}, Y_{max}) : Les valeurs maximales des coordonnées aux bords de l'image demeurent $(X_{max}, Y_{max}) = (1,1)$.
8. (L_x, L_y) : Le nombre de pixels dans l'image demeure aussi $(L_x, L_y) = (20, 20)$.

Nous résumons les changements dans les plages de paramètres dans le tableau suivant.

Tableau 5-1 : Changements aux plages de paramètres pour l'introduction des entées multiples

Paramètre	*Plage pour deux entrées*	*Plage pour n entrées*
Noyau	Carré, Triangle, Parabole, Vague	Triangle
n	2	$\mathbb{N}^+$
$\overrightarrow{W}$	$[-1, 1]$ ou $[0, 2]$	$[-1, 2]$
$\vec{S}$	$[0, 2{,}5]$	$[0, 2{,}5]$
$\overrightarrow{\Theta}$	$[0, 9 + 2s_x]$	$[0, 9 + 2s_1]$
Γ	1	1
(X_{max}, Y_{max})	$(1, 1)$	$(1, 1)$
(L_x, L_y)	$(20, 20)$	$(20, 20)$

En produisant des images à deux entrées, nous éliminons automatiquement les images si elles peuvent être créées par une entrée unique. Similairement pour $n \geq 2$ entrées, nous voulons éliminer les images qui peuvent être créées avec un nombre inférieur d'entrées. Par contre, il

n'est pas aussi facile d'en faire la détection uniquement par une règle mathématique applicable à l'image. Alternativement, nous avons modifié l'algorithme de la façon suivante : une fois l'image calculée pour le quantron avec toutes les entrées, chacune des entrées est retirée individuellement afin de créer une nouvelle image, celles-ci sont comparées à la première et dès qu'une de celles-ci est identique à l'originale, l'ensemble de paramètres est rejeté.

Nous avons ainsi conservé dans la banque seulement les images qui nécessitent bien toutes les entrées qu'elles utilisent. Par contre, pour chaque image à n entrées, nous devons en calculer $n + 1$ pour nous assurer ceci. De plus, le nombre de signaux à produire et à additionner passe également à n et le temps de calcul de chaque sortie est proportionnellement augmenté. Nous avons donc, par rapport au quantron de deux entrées exploré plus tôt, un temps de production $n(n + 1)$ fois plus long. Un nouveau problème s'ajoute du fait que le nombre d'images rejetées automatiquement s'avère beaucoup plus grand au fur et à mesure que n augmente. Pour contourner en partie ce problème, les images utilisées pour le test décrit ci-dessus sont d'abord produites avec une résolution de $L = 10$. La résolution $L = 20$ est seulement calculée si l'image n'est pas rejetée. Bien que ceci ajoute une image à calculer lorsque l'image est valide, le rapport des images rejetées contre celles acceptées est suffisamment grand pour que ce changement améliore la rapidité de la production par un facteur proche de 4.

Quelques centaines d'images ont été produites pour les quantrons « XXY » et « $XXYY$ ». La production d'images avec des entrées triples s'est avérée extrêmement lente, et les images peu différentes de celles produites avec les entrées doubles. Nous n'avons donc pas exploré plus loin la multiplication des entrées. Une nouvelle inspection des images ainsi produites nous a permis de former quelques nouvelles lettres.

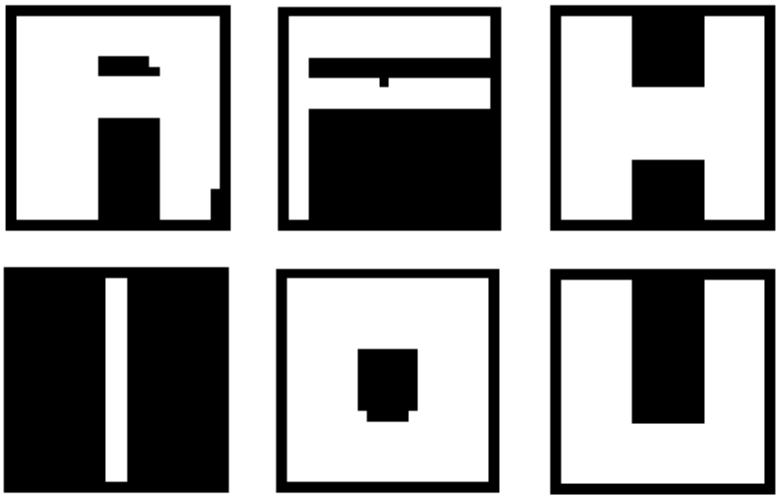

Figure 5.2 : Nouvelles images de lettres obtenues avec des quantrons utilisant plusieurs fois la même entrée

De nouveau, à titre d'exemple nous illustrons ici l'architecture et les paramètres du MLQ produisant la lettre *A*.

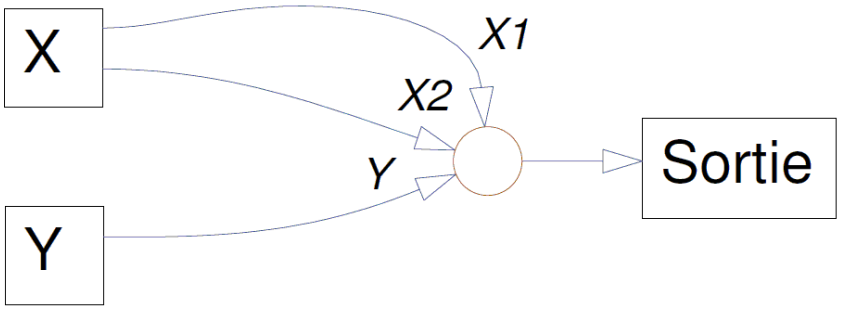

Figure 5.3 : Architecture du MLQ produisant la lettre *A* avec deux utilisation de la valeur d'entrée *X*

Ses paramètres sont :

$$\begin{pmatrix} w_{x1} \\ S_{x1} \\ \theta_{x1} \\ w_{x2} \\ S_{x2} \\ \theta_{x2} \\ w_y \\ S_y \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}5486 \\ 0{,}3484 \\ 7{,}6452 \\ 0{,}1829 \\ 1{,}3074 \\ 9{,}6786 \\ 0{,}3126 \\ 1{,}4122 \\ 0 \end{pmatrix} \tag{5.3}$$

Dans cette section, une nouvelle propriété du quantron découverte est expliquée. Cette propriété ajoute une nouvelle dimension au problème et peut avoir des répercussions importantes sur l'utilisation de MLQs dans le futur. Biologiquement, il est vrai qu'un neurone forme beaucoup plus qu'une seule connexion avec ses voisins (Kurzweil, 2005). Dans la prochaine section, nous étudions plus en profondeur les signaux qui forment les lettres obtenues jusque là.

6 ÉTUDE DES SIGNAUX

6.1 Objectifs et méthode

Nous avons déjà tenté d'améliorer l'aspect des lettres produites en modifiant au hasard leurs paramètres. Cette tentative a eu un succès partiel pour certaines lettres, mais nous n'en avons pas tiré beaucoup d'information concernant les rôles qu'ont les différents paramètres sur les modifications encourues aux images. Ceci est dû à la modification simultanée de tous les paramètres, rendant difficile l'observation de leurs effets individuels. L'avantage de la méthode était plutôt la production rapide et automatique d'images modifiées. Nous avons cette fois procédé à une amélioration manuelle des images. C'est-à-dire que pour chacune des images, nous modifions les paramètres individuellement afin d'observer leurs effets. La procédure est la suivante.

1. Nous choisissons une lettre à améliorer et obtenons les paramètres qui la produisent.
2. En maintenant les autres paramètres à leurs valeurs originales, nous modifions graduellement un des paramètres jusqu'à ce que nous observions une modification à l'image. La valeur du changement de paramètre requis pour obtenir la modification est notée Δp.
3. Nous produisons dix images qui représentent la transition du paramètre P par l'ensemble de valeurs $\{P - 4\Delta p, P - 3\Delta p, \cdots, P + 5\Delta p\}$.
4. Les étapes 2 et 3 sont répétées pour chacun de paramètres de l'image.
5. Les images obtenues sont rassemblées puis analysées. Nous notons alors les effets observables des différents paramètres.
6. Nous étudions les paramètres en fonction des signaux qu'ils produisent pour tenter de faire une correspondance avec les effets sur les images.
7. Nous modifions ensuite chaque paramètre en fonction de nos observations afin d'obtenir une image améliorée.

Lors de la dernière étape, il est parfois nécessaire de modifier les paramètres à plusieurs reprises afin d'obtenir le résultat final, ceux-ci ayant des effets interactifs que nous ne détectons pas avec l'analyse préalable. Ceci a été fait pour chaque lettre que nous désirons améliorer. Le but est

d'obtenir une meilleure compréhension des effets des paramètres afin de mieux maîtriser le quantron.

6.2 L, J, T, E, N, F, A, P

Nous présentons ici les différentes analyses effectuées, soient les lettres originales et leurs paramètres, les images des transitions, les observations pour chaque paramètre, les analyses des signaux, les modifications appliquées aux paramètres puis les lettres finales obtenues.

6.2.1 L

La première analysée de la sorte, nous avons choisie d'utiliser la lettre *L* pour sa forme simple obtenue à l'aide d'un seul quantron. L'image et les paramètres originaux sont les suivants.

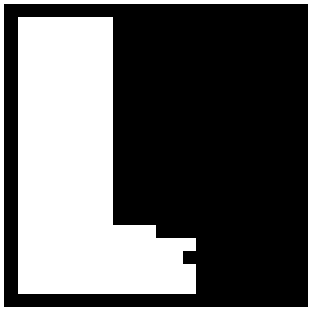

Figure 6.1 : La lettre *L* avant les améliorations

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}5829 \\ 0{,}3372 \\ 0{,}0115 \\ 1{,}0342 \\ 0 \\ 6{,}1019 \end{pmatrix} \tag{6.1}$$

Nous obtenons les images de transition suivantes.

Figure 6.2 : Effets des modifications de paramètres sur la lettre *L*

Nous observons dans l'ordre:

- w_x : ce paramètre fait allonger la patte inférieure du L. Il y a aussi une augmentation graduelle de son épaisseur à partir de la jonction avec la barre verticale.
- w_y : ce paramètre pousse l'ensemble de l'image vers la droite, il fait épaissir la barre verticale et repousse du même coup la patte inférieure, sans toutefois la faire allonger.
- S_x : ce paramètre fait allonger la patte inférieure du L mais contrairement à w_x elle ne la fait pas s'épaissir.
- S_y : ce paramètre fait épaissir la barre verticale, mais ne repousse pas autant la patte inférieure qui tend à se faire engloutir.
- θ_y : ce paramètre n'affecte pas la bande verticale, seulement l'épaisseur de la patte inférieure est diminuée.

Nous avons ici deux signaux aux poids positifs. Le signal X a une amplitude légèrement plus élevée que Y mais n'arrive pas par lui-même à atteindre le seuil, puisque ses noyaux sont très étroits (S_x petit), et ce même pour les fréquences les plus élevées. Il n'y a donc jamais

d'activation dans le haut de l'image, et sa contribution nécessite l'interaction avec le signal Y. Le signal Y, lui, arrive à atteindre le seuil de lui-même grâce à ses larges noyaux (S_y grand) ce qui produit la bande verticale à gauche. Accroître w_y ou S_y augmente le maximum de la superposition des noyaux du signal Y et rend le seuil atteignable à des fréquences plus élevées, d'où l'épaississement de la bande verticale. Nous avons une valeur initiale de $\theta_y \cong 6$ ce qui implique que le signal X ne réussira à atteindre le signal Y qu'à approximativement $X \cong \theta_y/N - 1 = 6/9 = 0.67$ soit les deux tiers de l'image. C'est à ce moment que le premier signal arrive à contribuer au second et que la patte inférieure apparaît. Il s'ensuit qu'augmenter la valeur de θ_y repousse la possibilité du contact et diminue l'épaisseur de la patte. Finalement, les paramètres $w_x, w_y, S_x,$ et S_y contribuent chacun à leur manière à augmenter l'amplitude de leurs signaux respectifs et donc du résultat de leur rencontre, fortifiant ainsi la présence de la patte inférieure.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}5829 + 0{,}15 \\ 0{,}3372 - 0{,}05 \\ 0{,}0115 + 0{,}03 \\ 1{,}0342 \\ 0 \\ 6{,}1019 \end{pmatrix} = \begin{pmatrix} 0{,}7329 \\ 0{,}2872 \\ 0{,}0415 \\ 1{,}0342 \\ 0 \\ 6{,}1019 \end{pmatrix} \tag{6.2}$$

Nous avons aminci la bande verticale en augmentant le paramètre w_y puis épaissi et allongé la patte inférieure en augmentant les paramètres w_x et S_x, ce qui produit l'image améliorée du L.

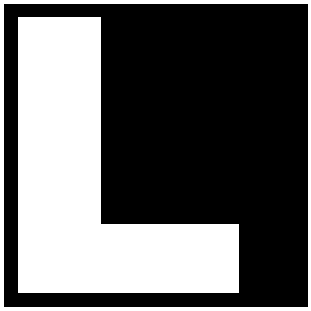

Figure 6.3 : La lettre L après les améliorations

La même démarche est utilisée pour les sept lettres qui suivent. Ayant déjà pris connaissance de l'exemple pour la lettre L, le lecteur peut désirer ne cibler que les lettres qu'il désire.

6.2.2 J

La seconde lettre améliorée est également produite par un seul quantron, par contre la lettre est un peu plus instable. Voici la lettre originale et ses paramètres, il est à noter que les couleurs sont inversées pour les images de transitions.

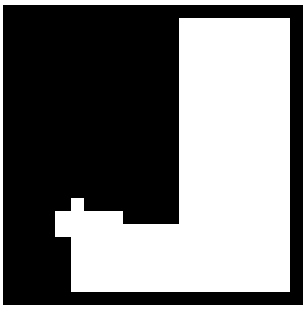

Figure 6.4 : La lettre J avant les améliorations

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} -0{,}6659 \\ 0{,}3588 \\ 1{,}8496 \\ 1{,}6346 \\ 0 \\ 5{,}0264 \end{pmatrix} \tag{6.3}$$

Nous obtenons les images de transition suivantes.

Figure 6.5 : Effets des modifications des paramètres sur la lettre J

Nous observons dans l'ordre :

- w_x : ce paramètre affecte la pointe du J, celle-ci prend de l'ampleur plus la valeur absolue augmente.
- w_y : ce paramètre affecte l'épaisseur de la barre verticale du J. Il repousse aussi la pointe et creuse la courbe.
- S_x : ce paramètre parvient à épaissir la patte du J sans trop modifier la longueur de la pointe.
- S_y : ce paramètre, comme w_y, affecte l'épaisseur de la barre verticale, mais affecte moins le creux de la courbe.
- θ_y : ce paramètre a un effet similaire à S_x mais dans le sens inverse.

Nous avons cette fois un signal positif et un signal négatif. Les deux valeurs S_x et S_y sont semblables alors les signaux individuels se comportent de manière similaire. Ces valeurs sont relativement grandes, toutes deux supérieures à 1,5. Considérant que la largeur totale d'un noyau est de $2S$, chaque noyau atteint au minimum le début de son troisième voisin, et ce, même pour la

valeur d'entrée maximale de 1. Ces valeurs de $\vec{S}$ produisent des signaux plutôt lisses avec un seul sommet. Le signal au poids négatif est aussi celui au délai nul et donc celui qui débute le premier. Puisque son maximum d'amplitude est généralement au centre du signal, celui-ci ne se retrouve jamais passé la valeur d'environ 5. Ainsi, considérant que le délai du signal positif est aussi de 5, le sommet de ce dernier n'arrive jamais avant le négatif. Nous savons que par lui-même, le signal Y positif arrive à atteindre le seuil pour au moins une partie de l'image, sinon l'image serait entièrement noire. Il y a activation dans l'image finale si, soit le sommet du signal Y est suffisamment élevé pour contrer l'effet négatif du signal X, ou s'il est suffisamment repoussé dans le temps pour en ressortir entièrement ou presque. Pour la partie supérieure de l'image, nous avons les valeurs les plus petites de l'entrée X. Ceci compacte ses noyaux et donne à sa résultante une grande amplitude, mais une courte étendue. La fin du signal négatif précède alors le début du signal positif qui est libre d'atteindre le seuil comme s'il était seul. C'est ce qui explique pourquoi, dans le haut de l'image, nous avons une simple bande verticale, ce que l'on obtiendrait avec une seule entrée. Dans la partie inférieure de l'image, la valeur du signal X augmente et son influence atteint le signal Y et l'atténue, ce qui créé la patte du J. À l'extrême gauche, les valeurs de Y sont suffisamment petites pour que le sommet de sa résultante parvienne malgré tout à contrecarrer l'atténuation et l'activation persiste. Pour un signal X pas encore trop étendu, lorsque Y augmente, son sommet parvient encore à quitter le signal négatif et active la sortie, c'est ce qui produit le creux de la patte. Finalement, avec un Y suffisamment petit, lorsque X augmente et que sa résultante s'étend un peu trop, son amplitude diminue et le signal positif parvient à nouveau à atteindre le seuil. C'est ce qui crée le retour d'activation sous la pointe du J.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} -0{,}6659 \\ 0{,}3588 + 0{,}06 \\ 1{,}8496 + 0{,}6 \\ 1{,}6346 \\ 0 \\ 5{,}0264 \end{pmatrix} = \begin{pmatrix} -0{,}6659 \\ 0{,}4188 \\ 2{,}4496 \\ 1{,}6346 \\ 0 \\ 5{,}0264 \end{pmatrix} \tag{6.4}$$

Ici nous avons aminci la bande verticale et augmenté l'ampleur du creux en augmentant w_{y1} puis nous avons raugmenté l'épaisseur de la patte en augmentant S_x, ce qui produit l'image améliorée du J :

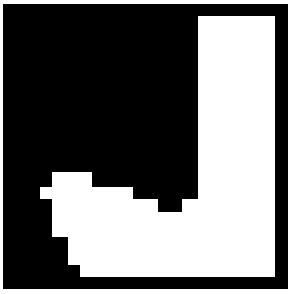

Figure 6.6 : La lettre J après les améliorations

6.2.3 T

L'image de lettre suivante est produite par une relation « ET » entre les sorties de deux quantrons de deux entrées. Les contributions individuelles sont plutôt simples et nous les analysons simultanément. Voici l'image originale et ses paramètres.

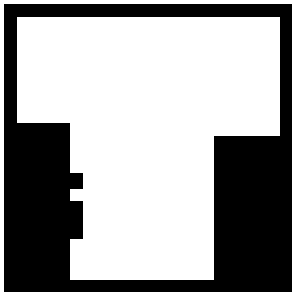

Figure 6.7 : La lettre T avant les améliorations

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} 0{,}4400 \\ 0{,}8122 \\ 1{,}0806 \\ 0{,}8784 \\ 0 \\ 10{,}4986 \end{pmatrix} \text{ et } \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 0{,}9381 \\ 0{,}0559 \\ 0{,}4557 \\ 1{,}9514 \\ 0{,}7448 \\ 0 \end{pmatrix} \quad (6.5)$$

Nous obtenons les images de transition suivantes.

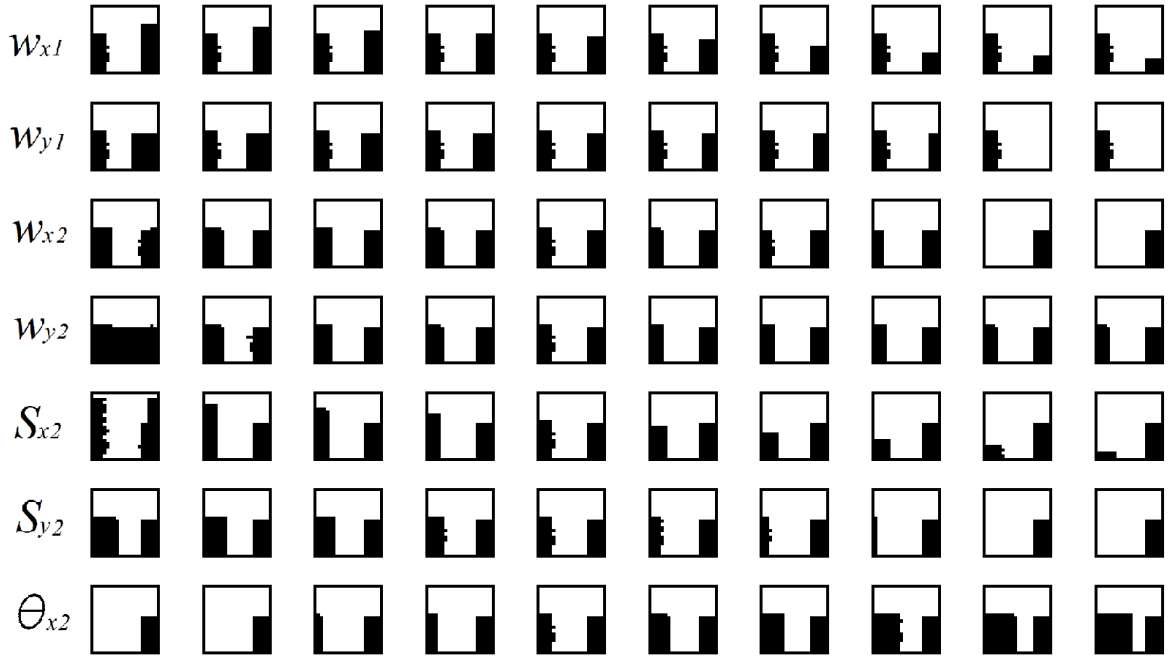

Figure 6.8 : Effets des modifications des paramètres sur la lettre T

Notons que nous ne montrons pas ici les transitions pour S_{x1}, S_{y1} et θ_{y1}. L'image du quantron 1 est en fait suffisamment simple pour pouvoir être modifiée uniquement avec les paramètres w_{x1} et w_{y1}.

Nous observons dans l'ordre :

- w_{x1} : ce paramètre affecte la hauteur de la bande noire de droite.
- w_{y1} : ce paramètre affecte l'épaisseur de la bande noire de droite.

- w_{x2} : ce paramètre affecte l'épaisseur de la bande noire de gauche, il affecte aussi légèrement sa hauteur, mais de façon moins prononcée.
- w_{y2} : ce paramètre affecte l'épaisseur de la bande noire de gauche.
- S_{x2} : ce paramètre affecte la hauteur de la bande noire de gauche.
- S_{y2} : ce paramètre affecte l'épaisseur de la bande noire de gauche.
- θ_{x2} : ce paramètre affecte l'épaisseur de la bande noire de gauche.

Le premier quantron affecte la partie droite de la lettre, soit la bande verticale noire qui s'y trouve. En fait, l'entrée X produit une activation tant qu'elle demeure sous une certaine valeur et l'entrée Y fait de même, créant ainsi des bandes blanches respectivement horizontale et verticale en haut et à gauche de l'image. La valeur élevée de θ_{y1} assure que les signaux ne se croisent jamais. Il s'agit en quelque sorte d'une relation « OU » entre deux quantrons d'une entrée. Les paramètres w_{x1} et w_{y1} peuvent alors être manipulés pour obtenir des bandes de largeurs variables.

Pour le deuxième quantron, nous avons deux poids positifs. Le poids de l'entrée X est très proche de 1 et permet d'atteindre le seuil facilement par lui-même et de produire une activation. Ceci fait apparaître une bande blanche verticale en haut de l'image. D'autre part, le poids de l'entrée Y est très petit et ne permet pas d'atteindre le seuil à lui seul. Par contre, lorsque l'entrée Y augmente, le signal réussit, à partir d'une certaine valeur, à atteindre le début du signal X décalé de θ_{x2} et à fournir ce qui manque à ce dernier pour atteindre le seuil. C'est ce qui produit alors la bande blanche verticale à droite de l'image. Augmenter w_{x2} ou w_{y2} augmente l'effet de l'interaction entre les signaux et permet à cette contribution d'atteindre le seuil plus rapidement, affectant ainsi l'épaisseur de la bande noire. Notons que w_{x2} affecte aussi légèrement la hauteur de la bande en permettant au signal X d'activer par lui-même la sortie plus ou moins rapidement, mais son effet précédent est plus prononcé. S_{x2} permet quand à lui d'ajuster cette hauteur sans affecter l'interaction entre les signaux. S_{y2} fait élargir les derniers noyaux du signal Y et lui permet d'atteindre le signal X plus rapidement, diminuant l'épaisseur de la bande. Le délais θ_{x2} décale directement le moment de la rencontre entre les signaux et fait épaissir la bande.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} 0{,}4400 - 0{,}15 \\ 0{,}8122 - 0{,}12 \\ 1{,}0806 \\ 0{,}8784 \\ 0 \\ 10{,}4986 \end{pmatrix} = \begin{pmatrix} 0{,}2900 \\ 0{,}6922 \\ 1{,}0806 \\ 0{,}8784 \\ 0 \\ 10{,}4986 \end{pmatrix}$$

$$\text{et} \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 0{,}9381 \\ 0{,}0559 \\ 0{,}4557 - 0{,}1 \\ 1{,}9514 - 0{,}6 \\ 4{,}7448 \\ 0 \end{pmatrix} = \begin{pmatrix} 0{,}9381 \\ 0{,}0559 \\ 0{,}3557 \\ 1{,}3514 \\ 4{,}7448 \\ 0 \end{pmatrix} \tag{6.6}$$

Nous avons désiré obtenir un T parfaitement symétrique avec les deux bandes blanches de la même épaisseur. Nous avons d'abord modifié w_{x1} et w_{y1} afin d'obtenir la bande noire de droite que nous désirions. Nous avons ensuite diminué S_{x2} et S_{y2} afin de respectivement rétrécir et amincir la bande noire de gauche. Ceci produit l'image améliorée du T.

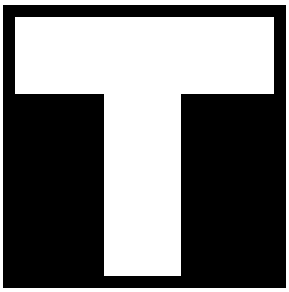

Figure 6.9 : La lettre T après les améliorations

6.2.4 E

Similairement à l'image de la lettre T, le E est produit par une relation « ET » entre les sorties de deux quantrons de deux entrées. Voici l'image originale et ses paramètres.

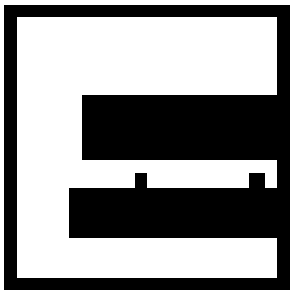

Figure 6.10 : La lettre E avant les améliorations

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} 0{,}8735 \\ 0{,}2436 \\ 0{,}7007 \\ 1{,}0184 \\ 0 \\ 8{,}2146 \end{pmatrix} \text{ et } \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 0{,}2031 \\ 0{,}9546 \\ 1{,}5907 \\ 0{,}2878 \\ 0 \\ 7{,}5957 \end{pmatrix} \tag{6.7}$$

Nous obtenons les images de transition suivantes respectivement pour le premier et deuxième quantron.

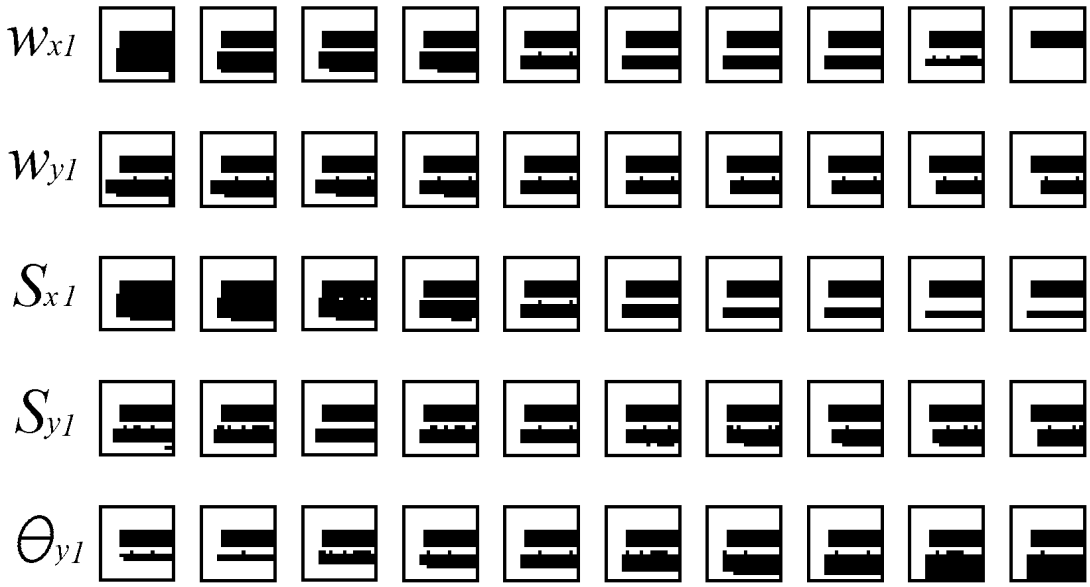

Figure 6.11 : Effets des modifications des paramètres du premier quantron sur la lettre E

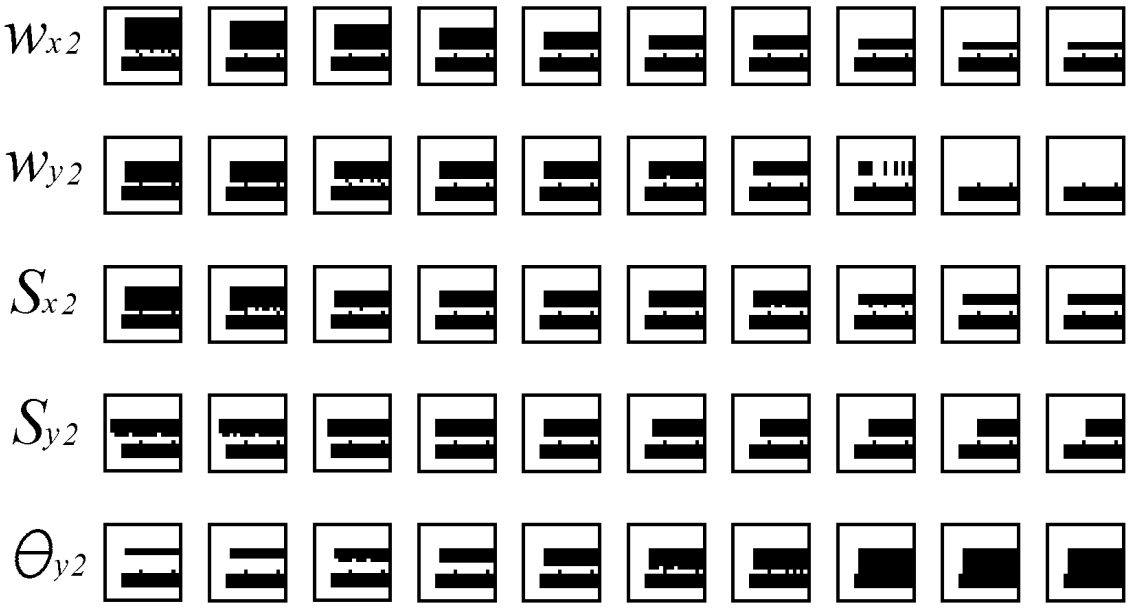

Figure 6.12 : Effets des modifications des paramètres du second quantron sur a lettre *E*

Notons déjà que chaque quantron contrôle une des bandes noires, le quantron 1 et le quantron 2 respectivement les bandes inférieure et supérieure. L'on s'attend à ce que le type d'interaction entre les signaux soit similaire dans les deux cas. Ceci est confirmé lorsque nous observons les effets des paramètres qui sont les mêmes sur leur bande respective. Nous allons donc noter les observations une seule fois.

Nous observons dans l'ordre :

- w_x : ce paramètre amincit la bande noire en augmentant les épaisseurs des bandes blanches en haut et en bas de l'image. La bande blanche du haut est la plus affectée.
- w_y : ce paramètre fait rétrécir la bande noire sur sa longueur en augmentant l'épaisseur de la bande blanche de gauche.
- S_x : ce paramètre a un effet presque identique à W_x.
- S_y : ce paramètre a un effet presque identique à W_y.
- θ_y : ce paramètre abaisse la limite inférieure de la bande noire en diminuant l'épaisseur de la bande blanche du bas.

Nous avons dans les deux cas des signaux X et Y positifs et capables par eux-mêmes de provoquer l'activation de la sortie lorsque leur entrée respective est suffisamment petite. Ceci provoque les bandes verticale et horizontale à gauche et en haut de l'image. À partir d'une certaine valeur de l'entrée X, le signal est trop étendu pour atteindre le seuil et cesse d'activer la sortie, ce qui termine la bande du haut. Par contre, alors que l'entrée augmente et qu'il s'étend d'avantage, il arrive à un certain point qu'il atteint le signal Y décalé de θ_y puis s'additionne à celui-ci pour créer l'activation de la sortie. Le type d'interaction est similaire à ce que nous avions pour le deuxième quantron de la lettre T, la différence étant qu'ici, les deux signaux parviennent au seuil par eux-mêmes. Nous avons donc w_x et S_x qui contribuent surtout à l'atteinte du seuil par le signal X et affectent donc la bande blanche du haut. Ils aident aussi à la superposition des deux signaux qui créent la bande inférieure, mais de façon moins prononcée. w_y et S_y ont un effet semblable, mais sur le signal Y et contribuent donc à l'épaisseur de la bande blanche de gauche. Finalement, θ_y est le principal responsable du moment de la rencontre entre les deux signaux et contrôle efficacement l'épaisseur de la bande blanche inférieure.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} 0{,}8735 \\ 0{,}2436 \\ 0{,}7007 - 0{,}03 \\ 1{,}0184 \\ 0 \\ 8{,}2146 - 0.6 \end{pmatrix} = \begin{pmatrix} 0{,}8735 \\ 0{,}2436 \\ 0{,}6707 \\ 1{,}0184 \\ 0 \\ 7{,}6146 \end{pmatrix}$$

$$\text{et} \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 0{,}2031 \\ 0{,}9546 \\ 1{,}5907 - 0{,}4 \\ 0{,}2878 - 0{,}06 \\ 0 \\ 7{,}5957 - 2 \end{pmatrix} = \begin{pmatrix} 0{,}2031 \\ 0{,}9546 \\ 1{,}1907 \\ 0{,}2278 \\ 0 \\ 5{,}5957 \end{pmatrix} \tag{6.8}$$

Nous avons désiré obtenir un E aux quatre bandes blanches de même épaisseur. Nous avons remonté la bande noire du haut en diminuant S_{y2} et θ_{y2} puis l'avons allongée en diminuant S_{y2}.

Nous avons ensuite remonté la bande noire du bas en diminuant S_{y1} et θ_{y1}. Ceci produit l'image améliorée du E :

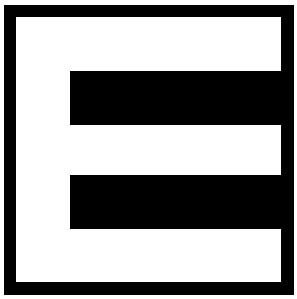

Figure 6.13 : La lettre E après les améliorations

6.2.5 N

La prochaine lettre est semblable aux deux dernières dans le sens où elle est formée par une relation « ET » entre deux quantrons. Le premier quantron contrôle la partie du bas et le second la partie haute du N. Voici l'image originale et ses paramètres.

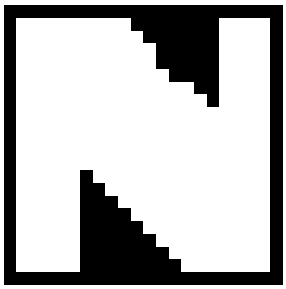

Figure 6.14 : La lettre N avant les améliorations

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} -0{,}5565 \\ 1{,}6200 \\ 1{,}0997 \\ 0{,}2764 \\ 0 \\ 3{,}7790 \end{pmatrix} \text{ et } \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 1{,}9358 \\ -0{,}4804 \\ 0{,}0140 \\ 1{,}0706 \\ 4{,}9691 \\ 0 \end{pmatrix} \tag{6.9}$$

Nous obtenons les images de transition suivantes pour le premier quantron.

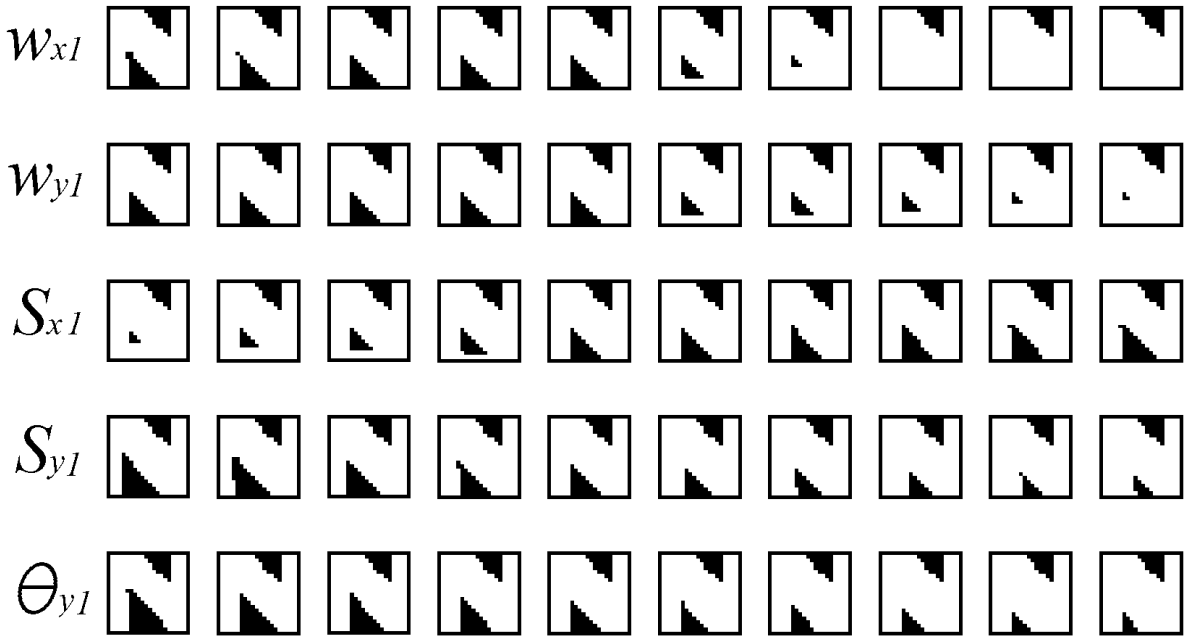

Figure 6.15 : Effets des modifications des paramètres sur le premier quantron de la lettre *N*

Nous observons dans l'ordre :

- w_{x1} : ce paramètre fait apparaître et amplifier une bande blanche horizontale en bas de l'image.
- w_{y1} : ce paramètre fait aussi apparaître et amplifier une bande blanche horizontale en bas de l'image.
- S_{x1} : ce paramètre fait aussi apparaître et amplifier une bande blanche horizontale en bas de l'image.
- S_{y1} : ce paramètre fait élargir la bande blanche verticale à gauche de l'image.
- θ_{y1} : ce paramètre fait augmenter la taille du triangle noir en repoussant sa limite diagonale.

Ensuite, pour le deuxième quantron nous obtenons les transitions qui suivent.

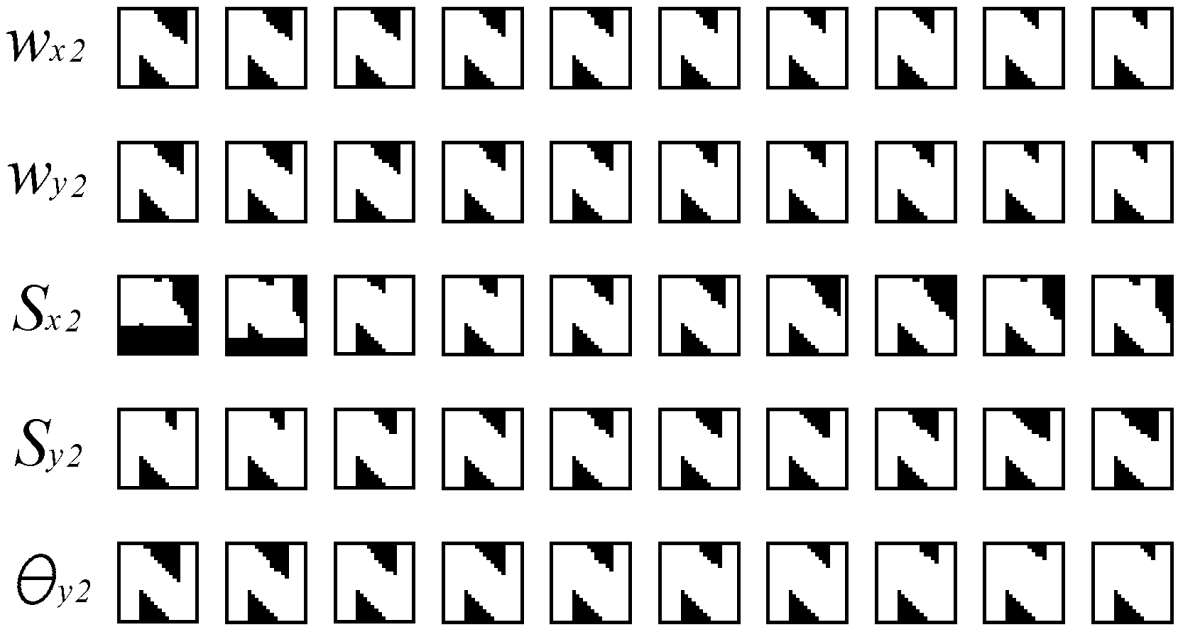

Figure 6.16 : Effets des modifications des paramètres sur le deuxième quantron de la lettre N

Nous observons dans l'ordre :

- w_{x2} : ce paramètre fait élargir la bande blanche verticale de droite.
- w_{y2} : ce paramètre fait aussi élargir la bande blanche verticale de droite.
- S_{x2} : ce paramètre affecte aussi la bande blanche verticale de droite, une modification plus prononcée produit des résultats étranges.
- S_{y2} : ce paramètre fait grossir le triangle vers la gauche et la droite.
- θ_{y2} : ce paramètre fait augmenter la taille du triangle noir en repoussant sa limite diagonale.

Pour le premier quantron, nous avons un poids positif supérieur à un pour l'entrée Y. Ceci signifie que le signal Y arrive à atteindre le seuil par lui-même, et ce, même sans superposition de ses noyaux et donc pour toute l'image. Par contre, le signal X est négatif et parvient à atténuer suffisamment le signal Y sur une partie de l'image. Pour que ceci se produise, l'entièreté des N noyaux du signal Y doit être atténuée d'au moins 0,6200. Le signal X débutant le premier, il doit

donc être plus étendu que le signal Y afin de le couvrir entièrement. C'est pourquoi nous obtenons le manque d'activation seulement en bas à gauche. La taille des noyaux du signal Y est relativement petite, mais permet tout de même pour une entrée assez réduite d'obtenir une superposition, le signal négatif est alors incapable d'atténuer suffisamment et nous obtenons la barre blanche verticale de gauche. À l'opposé, lorsque l'entrée Y augmente, le dernier noyau est repoussé hors de la portée de la fin du signal X et l'activation est retrouvée. Alors que l'entrée X augmente, le signal rattrape le dernier noyau Y et réussit à l'atténuer, c'est cette course réciproque qui provoque l'hypoténuse du triangle noir du bas. Ainsi, toute modification de paramètre qui favorise le signal Y fera disparaître graduellement le triangle. Particulièrement, augmenter la taille des noyaux positifs permet d'obtenir une superposition plus facilement et élargit la bande de gauche. Plus utile est l'effet du délais θ_{y1} qui permet de modifier la limite où le dernier sommet positif échappe à l'atténuation et modifie donc la limite oblique du triangle.

Une situation semblable se produit pour le second quantron qui contrôle le triangle noir du haut. Le signal positif est encore une fois suffisant à lui seul pour activer la sortie et ses N noyaux doivent être atténués pour obtenir un pixel noir. Par contre, les noyaux positifs sont suffisamment minces (S_{x2} petit) pour qu'ils ne se superposent pas et nous n'avons donc pas de bande blanche horizontale en haut de l'image. La bande verticale à droite s'obtient quant à elle lorsque le signal négatif Y est soudainement trop étendu et bien qu'il couvre tous les noyaux de X, son amplitude n'est plus assez élevée pour annuler l'activation. Encore une fois, modifier un paramètre qui favorise le signal positif fera élargir la bande verticale de droite. Augmenter la taille des noyaux négatifs permet une couverture plus rapide du dernier noyau positif et augmente ainsi légèrement la taille du triangle vers la gauche, mais le moyen le plus efficace d'obtenir ce résultat est avec θ_{x2}. Notons encore le comportement étrange produit par la modification de S_{x2}. Ceci est en fait dû à sa faible valeur. Nous verrons en quoi cette caractéristique est conséquente lors de l'analyse de la précision $nbpts$.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} -0{,}5565 \\ 1{,}6200 - 0.02 \\ 1{,}0997 \\ 0{,}2764 + 0.07 \\ 0 \\ 3{,}7790 - 0.43 \end{pmatrix} = \begin{pmatrix} -0{,}5565 \\ 1{,}6000 \\ 1{,}0997 \\ 0{,}3464 \\ 0 \\ 3{,}3490 \end{pmatrix}$$

$$\text{et} \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 1{,}9358 \\ -0{,}4804 \\ 0{,}014 - 0{,}0035 \\ 1{,}0706 \\ 4{,}9691 - 1{,}2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1{,}9358 \\ -0{,}4804 \\ 0{,}0105 \\ 1{,}0706 \\ 3{,}7691 \\ 0 \end{pmatrix} \quad (6.10)$$

Nous avons désiré obtenir un N avec les bandes verticale et diagonale de mêmes grosseurs, avec la bande diagonale bien centrée. Pour ce faire, les triangles noirs se devaient d'être identiques. Nous avons donc modifié S_{y1} et S_{x2} pour ajuster leurs côtés droits puis θ_{y1} et θ_{x2} pour leurs hypoténuses. Le poids w_{y1} est légèrement diminué pour éliminer une bande indésirable au bas de l'image. Ceci produit l'image améliorée du N.

Figure 6.17 : La lettre N après les améliorations

6.2.6 F

La lettre suivante ne requiert qu'un seul quantron de deux entrées pour être produite. Par contre, le résultat est à première vue inattendu et c'est une image que nous n'avons obtenue qu'une seule

fois lors de la production du grand nombre d'images aléatoires. Nous nous attendons donc à ce qu'elle soit relativement instable et c'est pourquoi nous ne l'avons pas analysée plus tôt. Voici l'image originale et ses paramètres.

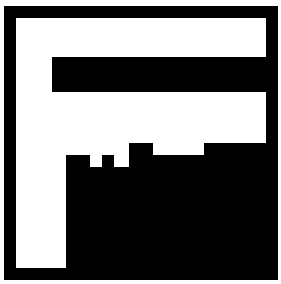

Figure 6.18 : La lettre F avant les améliorations

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}2002 \\ 0{,}6973 \\ 0{,}8352 \\ 0{,}2332 \\ 0 \\ 3{,}7413 \end{pmatrix} \tag{6.11}$$

Nous obtenons les images de transition suivantes.

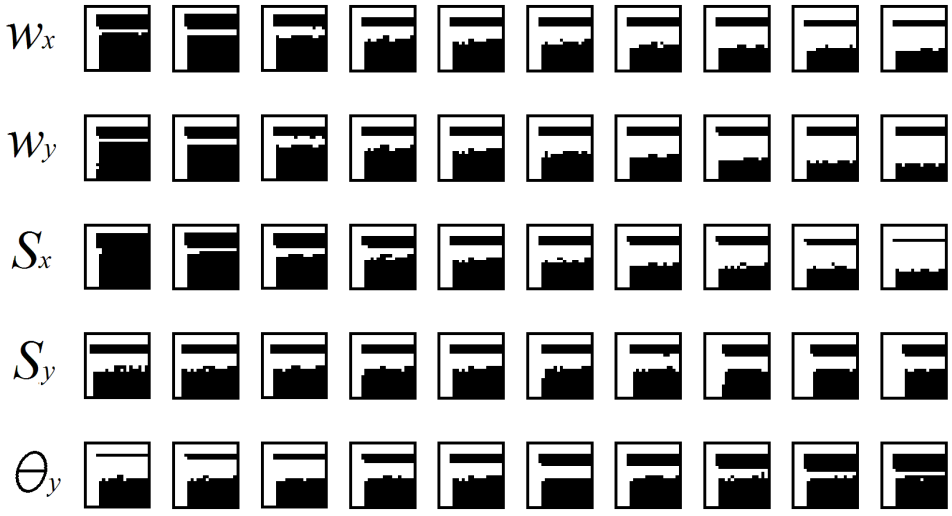

Figure 6.19 : Effets des modifications des paramètres sur la lettre F

Nous observons dans l'ordre :

- w_x : ce paramètre permet d'épaissir les deux bandes blanches horizontales, particulièrement celle du bas.
- w_y : ce paramètre permet aussi d'épaissir la bande blanche horizontale du bas, mais sans affecter celle du haut.
- S_x : ce paramètre a le même effet que w_x.
- S_y : ce paramètre fait épaissir la barre blanche verticale à gauche.
- θ_y : ce paramètre abaisse la partie supérieure de la barre horizontale du bas.

Nous avons ici deux poids positifs. À eux-mêmes, les signaux arrivent à provoquer l'activation de la sortie lorsque leurs entrées respectives sont suffisamment petites. w_x est plus grand que w_y mais cette différence est compensée par l'écart équivalent entre S_x et S_y. Il en résulte des barres horizontale et verticale respectivement en haut et à gauche de l'image de tailles semblables. La valeur relativement élevée S_x de la taille des noyaux du signal X assure que ceux-ci se superposent au moins partiellement dans toute l'image. Alors que l'entrée X augmente, le signal diminue en amplitude et n'atteint plus le seuil, la bande horizontale du haut cesse. Par contre, celui-ci s'allonge et parvient à rejoindre le premier noyau du signal Y. La petite taille S_y des noyaux du signal Y fait en sorte qu'ils sont rapidement isolés de leurs voisins, ce totalement à partir de $Y = 2S_y \cong 0{,}47$ et donc à la moitié de l'image. En fait, le sommet du premier noyau ne se voit plus aidé du noyau suivant à partir de $Y = S_y \cong 0{,}23$ et donc le quart de l'image. Si la somme de ce premier noyau et du signal étiré X est suffisante pour atteindre le seuil, nous obtenons l'activation et c'est ce qui produit le début de la deuxième bande horizontale. Alors que l'entrée X continue d'augmenter en descendant dans l'image, le signal s'étire davantage et diminue en amplitude jusqu'à ce que la somme de celui-ci et du premier noyau Y ne parvienne plus au seuil et l'activation est perdue, c'est ce qui provoque la fin de la bande. Alors que les deux entrées continuent d'augmenter, le maximum du signal Y demeure à w_y et le signal X qui continue à s'étirer ne peut plus aspirer à contribuer suffisamment pour atteindre le seuil même s'il atteint les noyaux suivants de Y. Les signaux sont alors incapables d'atteindre le seuil par eux-mêmes ou par leur interaction et le reste de l'image est noir. Ainsi w_x et S_x contribuent

naturellement à la taille de la barre horizontale du haut, mais aussi à prolonger la limite inférieure de la bande du bas qui provient de l'interaction XY. w_y a le même effet sur l'interaction et donc la barre inférieure, mais sans affecter la barre du haut qui ne dépend que du signal X lorsqu'il est isolé. Il devrait aussi affecter la taille de la bande verticale, mais son effet sur l'interaction est trop fort pour l'observer sur les images de transition produites. S_y contribue normalement à la barre verticale de gauche, mais son augmentation faible ne modifie pas l'isolation de ses noyaux dans les trois quarts de l'image. θ_y permet comme toujours de repousser le moment ou l'interaction se produit et donc la barre horizontale du bas qui en résulte. Notons ici la présence de pixels isolés particulièrement en bas de la deuxième bande horizontale. Nous attribuons ceci à la précision $nbpts$ dont nous discuterons plus tard.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}2002 + 0{,}05 \\ 0{,}6973 - 0{,}025 \\ 0{,}8352 \\ 0{,}2332 + 0{,}045 \\ 0 \\ 3{,}7413 + 0{,}9 \end{pmatrix} = \begin{pmatrix} 0{,}2502 \\ 0{,}6723 \\ 0{,}8352 \\ 0{,}2782 \\ 0 \\ 4{,}6413 \end{pmatrix}. \tag{6.12}$$

Nous avons autant que possible tenté d'obtenir un F identique au E obtenu auparavant, mais sans la dernière bande blanche horizontale. Nous avons d'abord ajusté l'épaisseur de la première bande horizontale à l'aide de w_x, de la seconde avec w_y puis de la bande verticale avec S_y. Nous avons ensuite abaissé le haut de la seconde bande avec θ_y. Il persiste une imperfection environ au centre de l'image que nous n'avons pas réussi à éliminer. L'image étant entièrement produite par un seul quantron, il est plus difficile d'obtenir des ajustements fins avec le nombre limité de paramètres. Voici l'image améliorée du F obtenue.

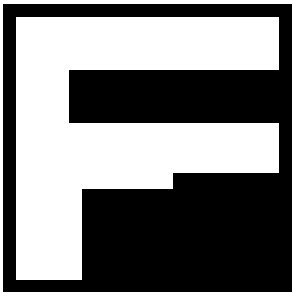

Figure 6.20 : La lettre F après les améliorations

6.2.7 A

La prochaine lettre requiert l'assemblage de trois quantrons de deux entrées. La première image forme le premier triangle qui sert de support à la lettre. Ce triangle est d'abord noir et doit être inversé pour obtenir une lettre blanche. Les deux images suivantes sont des blocs noirs qui produisent les trous dans le triangle de départ et forment ainsi la lettre A. Voici l'image originale et ses paramètres.

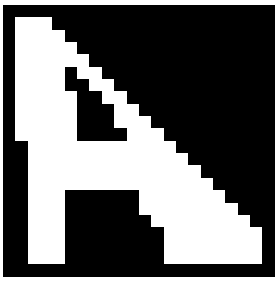

Figure 6.21 : La lettre A avant les améliorations

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} -0{,}7548 \\ 1{,}0958 \\ 1{,}6341 \\ 0{,}1266 \\ 0 \\ 1{,}7071 \end{pmatrix}, \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 0{,}4903 \\ 0{,}1416 \\ 1{,}4500 \\ 1{,}7707 \\ 5{,}4834 \\ 0 \end{pmatrix} \text{ et } \begin{pmatrix} w_{x3} \\ w_{y3} \\ S_{x3} \\ S_{y3} \\ \theta_{x3} \\ \theta_{y3} \end{pmatrix} = \begin{pmatrix} -0.3958 \\ 1{,}9577 \\ 1{,}8614 \\ 0{,}3033 \\ 0 \\ 1{,}7022 \end{pmatrix} \quad (6.13)$$

Notons d'abord la similitude dans les paramètres du quantron 1 et 3. Les deux quantrons produisent en fait des images similaires, soient des triangles presque rectangles à l'angle carré en bas à gauche. Nous avons déjà utilisé ce type de forme lors de la création de la lettre N. Nous remarquons d'ailleurs approximativement les mêmes proportions des paramètres dans le quantron 1 de cette lettre. Les trois quantrons produisent en fait des interactions similaires entre leurs signaux et font partie de ce que nous appelons une catégorie. Nous pouvons utiliser les mêmes remarques faites lors de l'analyse du N pour ces deux quantrons et nous ne les répétons pas ici. Nous produisons toutefois les images de transition pour le deuxième quantron, voici les résultats.

Figure 6.22 : Effets des modifications des paramètres sur le deuxième quantron de la lettre A

Nous observons dans l'ordre :

- W_{x2} : ce paramètre fait abaisser le bloc et reculer sa limite droite.
- W_{y2} : ce paramètre amincit le bloc à gauche et à droite.
- S_{x2} : ce paramètre a un effet similaire à W_{x2}.
- S_{y2} : ce paramètre a un effet similaire à W_{y2}.
- θ_{x2} : ce paramètre épaissit le bloc en faisant avancer uniquement la partie droite.

L'interaction entre les signaux n'est pas très complexe. Les deux signaux sont positifs et leurs larges noyaux créent des courbes relativement lisses et uniformes. Les deux signaux arrivent à atteindre le seuil pour des petites valeurs d'entrées, le poids w_x est plus grand que le poids w_y alors la bande du haut est plus épaisse que celle de gauche. C'est θ_{x2} qui est non-nul, alors le signal Y est le premier actif. Lorsqu'il atteint le sommet du signal X il parvient à l'aider à atteindre le seuil, ce qui crée l'activation à droite de du bloc noir. Par contre, ce sommet est repoussé lorsque l'entrée X augmente, alors la limite du bloc obtenue est diagonale. Ainsi les paramètres w_{x2}, w_{y2}, S_{x2} et S_{y2} affectent les bandes horizontales et verticales selon leurs entrées respectives et contribuent tous à l'interaction, affectant la limite diagonale, puis θ_{x2} repousse cette même diagonale en repoussant le moment de l'interaction.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_{x1} \\ w_{y1} \\ S_{x1} \\ S_{y1} \\ \theta_{x1} \\ \theta_{y1} \end{pmatrix} = \begin{pmatrix} -0{,}7548 + 0{,}69 \\ 1{,}0958 - 0{,}015 \\ 1{,}6341 + 0{,}25 \\ 0{,}1266 \\ 0 \\ 1{,}7071 - 0{,}2 \end{pmatrix} = \begin{pmatrix} -0{,}0648 \\ 1{,}0808 \\ 1{,}8841 \\ 0{,}1266 \\ 0 \\ 1{,}5071 \end{pmatrix}, \begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} 0{,}4903 \\ 0{,}1416 + 0{,}025 \\ 1{,}4500 \\ 1{,}7707 \\ 5{,}4834 + 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0{,}4903 \\ 0{,}1666 \\ 1{,}4500 \\ 1{,}7707 \\ 6{,}4834 \\ 0 \end{pmatrix}$$

$$\text{et } \begin{pmatrix} w_{x3} \\ w_{y3} \\ S_{x3} \\ S_{y3} \\ \theta_{x3} \\ \theta_{y3} \end{pmatrix} = \begin{pmatrix} -0{,}3958 \\ 1{,}9577 \\ 1{,}8614 \\ 0{,}3033 + 0{,}03 \\ 0 \\ 1{,}7022 + 0{,}1 \end{pmatrix} = \begin{pmatrix} -0{,}3958 \\ 1{,}9577 \\ 1{,}8614 \\ 0{,}3333 \\ 0 \\ 1{,}8022 \end{pmatrix} \tag{6.14}$$

Nous avons tenté d'obtenir un A aux bandes blanches approximativement de mêmes épaisseurs. La patte gauche ne peut malheureusement être que verticale, nous avons au moins réussi à la rendre parfaitement droite. Nous obtenons par contre une limite oblique à l'extrême droite de la lettre légèrement moins belle qu'avant. Une fois la base de la lettre obtenue, nous avons modifié les deux autres blocs noirs afin de compléter l'intérieur de la lettre le mieux possible. Voici l'image améliorée du A obtenue:

Figure 6.23 : La lettre A après les améliorations

6.2.8 P

Nous avons ici une image créée par un seul quantron, mais avec trois entrées XXY. L'image obtenue peut être considérée telle quelle comme la lettre A majuscule, mais le A est plutôt carré puisqu'il ne possède pas de pattes obliques. Nous nous rendons compte en modifiant les paramètres qu'il est aussi possible d'obtenir la lettre P, bien que carrée elle aussi. Voici l'image originale et ses paramètres.

Figure 6.24 : La seconde lettre A avant les modifications

$$\begin{pmatrix} w_{x1} \\ S_{x1} \\ \theta_{x1} \\ w_{x2} \\ S_{x2} \\ \theta_{x2} \\ w_y \\ S_y \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}5486 \\ 0{,}3484 \\ 7{,}6452 \\ 0{,}1829 \\ 1{,}3074 \\ 9{,}6786 \\ 0{,}3126 \\ 1{,}4122 \\ 0 \end{pmatrix} \tag{6.15}$$

Nous obtenons les images de transition suivantes.

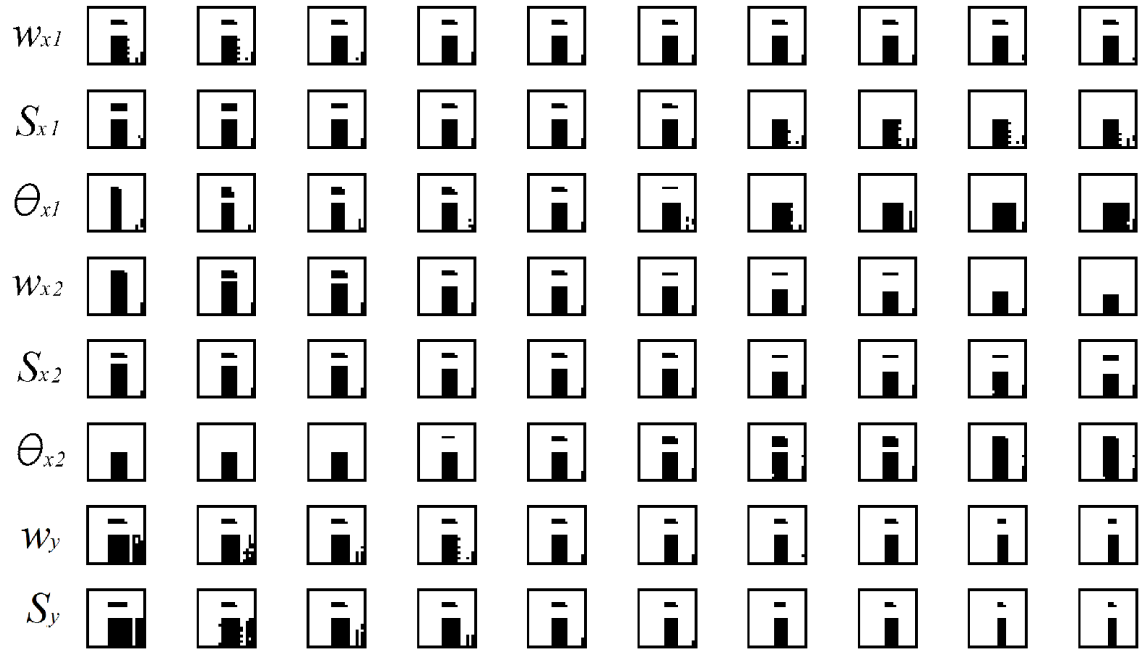

Figure 6.25 : Effets des modifications des paramètres sur la seconde lettre *A*

Nous observons dans l'ordre :

- w_{x1} : ce paramètre permet d'éliminer les pixels « bruités » sur la bande droite de l'image.
- S_{x1} : ce paramètre fait disparaître le rectangle noir du haut et affecte les pixels bruités.
- θ_{x1} : ce paramètre fait épaissir la bade blanche horizontale du centre et amincit la bande verticale de droite.
- w_{x2} : ce paramètre permet d'augmenter l'épaisseur de la bande horizontale du centre et légèrement celle du haut.
- S_{x2} : ce paramètre a un effet similaire à w_{x2}.
- θ_{x2} : ce paramètre a un effet similaire à w_{x2} et S_{x2} mais dans le sens opposé.
- w_y : ce paramètre affecte les deux bandes verticales, particulièrement dans la partie du bas de la bande de droite.
- S_y : ce paramètre a un effet similaire à w_y.

Nous avons cette fois trois signaux positifs qui s'entrecoupent pour former l'image. En ordre croissant, les délais sont $\theta_y < \theta_{x1} < \theta_{x2}$, ainsi le premier signal est Y, suivit de X_1 et X_2. Le signal Y arrive au seuil par lui-même pour ses petites valeurs d'entrée et est le seul responsable de la bande verticale de gauche. Les deux signaux en X parviennent aussi au seuil, mais le second persiste plus longtemps et c'est lui qui contrôle la bande du haut. Le délai « effectif » entre les signaux X : $\theta_e = \theta_{x2} - \theta_{x1} = 9{,}6786 - 7{,}6452 = 2{,}0334$ est relativement petit et les signaux arrivent rapidement à interagir pendant qu'ils ont une amplitude suffisante. Ceci produit le début de la bande horizontale du centre, par contre cette superposition ne perdure pas et la bande s'arrête avant la fin de l'image. La distance entre le signal Y et les deux autres est comparativement grande, la distance à parcourir pour réussir à contribuer fait en sorte que l'effet est obtenu plus loin dans l'image, soit la bande verticale de droite. La partie du bas de cette bande tend par contre à disparaître, car les signaux X sont alors de plus faibles amplitudes. Ainsi, les paramètres w_{x1}, S_{x1}, w_{x2} et S_{x2} contribuent tous à l'interaction entre les signaux X et affectent donc la bande du centre. w_{x1} et S_{x1} affectent le premier signal X qui sera rencontré par le signal Y et aident à fortifier le bas de la bande de droite. θ_{x1} et θ_{x2} influencent le délai effectif entre les deux signaux X et décalent donc le moment de l'apparition de la bande du centre. θ_{x1} contrôle aussi le moment de l'apparition de la bande de gauche. w_y et S_y eux contrôlent naturellement la bande de gauche dont ils sont les seuls responsables, mais aussi l'interaction qui produit la bande de droite.

Nous arrivons à modifier les paramètres pour obtenir le nouvel ensemble suivant.

$$\begin{pmatrix} w_{x1} \\ S_{x1} \\ \theta_{x1} \\ w_{x2} \\ S_{x2} \\ \theta_{x2} \\ w_y \\ S_y \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}5486 \\ 0{,}3484 \\ 7{,}6452 - 0{,}25 \\ 0{,}1829 \\ 1{,}3074 \\ 9{,}6786 + 0{,}4 \\ 0{,}3126 \\ 1{,}4122 + 0{,}1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0{,}5486 \\ 0{,}3484 \\ 7{,}3952 \\ 0{,}1829 \\ 1{,}3074 \\ 10{,}0786 \\ 0{,}3126 \\ 1{,}5122 \\ 0 \end{pmatrix} \tag{6.16}$$

Nous avons tenté d'obtenir un A symétrique par rapport à sont axe vertical avec des bandes verticales de mêmes largeurs du haut au bas de l'image. Nous avons d'abord augmenté S_y afin d'éliminer les pixels noirs dans le coin bas droit. Nous avons ensuite diminué θ_{x1} afin d'épaissir le bas de la bande verticale de droite. Finalement, nous avons augmenté θ_{x2} pour repousser la bande du centre et du coup augmenter la taille du carré noir. Voici l'image améliorée du A obtenue.

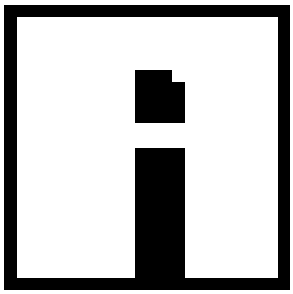

Figure 6.26 : La lettre A après les améliorations

Nous avons aussi remarqué qu'il était possible d'obtenir une nouvelle lettre en ne modifiant que légèrement les paramètres de départ. En effet, nous obtenons la lettre P avec les paramètres suivants.

$$\begin{pmatrix} w_{x1} \\ S_{x1} \\ \theta_{x1} \\ w_{x2} \\ S_{x2} \\ \theta_{x2} \\ w_y \\ S_y \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}5486 \\ 0{,}3484 \\ 7{,}6452 \\ 0{,}1829 \\ 1{,}3074 \\ 9{,}6786 \\ 0{,}3126 \\ 1{,}4122 - 0{,}32 \\ 0 \end{pmatrix} = \begin{pmatrix} 0{,}5486 \\ 0{,}3484 \\ 7{,}6452 \\ 0{,}1829 \\ 1{,}3074 \\ 9{,}6786 \\ 0{,}3126 \\ 1{,}0922 \\ 0 \end{pmatrix} \tag{6.17}$$

Nous n'avons que diminué la valeur de S_y suffisamment pour atteindre le point où la partie du bas de la bande de droite disparaît. Voici l'image du P obtenue.

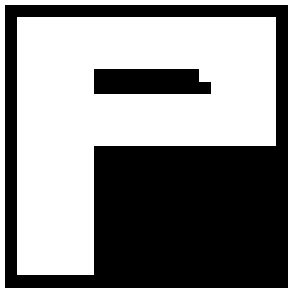

Figure 6.27 : La lettre *P* après les améliorations

6.3 Résumé

Nous avons, pour chacune des lettres obtenues, étudié et analysé les signaux qui les produisent et les paramètres qui en sont responsables. Ceci nous a permis tout d'abord de les modifier de façon éclairée afin d'améliorer l'aspect des lettres. Nous avons aussi maintenant une meilleure compréhension des interactions possibles entre les signaux qui produisent les surfaces discriminantes que nous observons. Nous nous servons de ces connaissances dans les sections suivantes afin d'obtenir d'autres résultats et propriétés du quantron. Les deux prochaines sections traitent des approximations imposées par la numérisation du modèle et de leurs conséquences.

7 PRÉCISION DE L'APPROXIMATION

Nous avons fait mention lors de la section « Implémentation du quantron sur Matlab » que nous avons un paramètre « $nbpts$ » contrôlant la précision de l'approximation des signaux qui déterminent la sortie du quantron. Nous détaillons ici le rôle exact de ce paramètre et nous montrons les tests des effets que nous avons effectués sur les images produites.

7.1 Vecteurs discrets

Le modèle du quantron décrit des signaux composés de noyaux qui se superposent sur un support de temps continu. Nous devons, pour ce travail, approximer ce modèle en échantillonnant le support qui devient discret. Nous calculons d'abord quelle est la valeur du temps maximal t_{max} où il persiste au moins un signal non nul selon la formule :

$$t_{max} = max\{\theta_k + (N-1)X_k + 2S_k\} \text{ pour } k = 1 \text{ à } n. \tag{7.1}$$

Ceci détermine la borne supérieure du support temps au-delà de laquelle il est inutile de continuer le calcul puisque tous les signaux sont alors nuls. Nous créons ensuite un vecteur d'une longueur de $nbpts$ où chaque élément représente un incrément d'une valeur de

$$\Delta t = t_{max}/(nbpts - 1). \tag{7.2}$$

Le vecteur débute ainsi à zéro et termine bien à t_{max} avec $nbpts$ éléments séparés par Δt. Cette valeur d'espacement entre les éléments Δt doit aussi être respectée lors de la création du vecteur qui forme le noyau et celui qui forme le peigne de Dirac afin de pouvoir effectuer la convolution discrète. Pour le noyau, les valeurs du vecteur sont calculées dans l'intervalle $[0,2S]$ pour chaque saut de Δt selon l'équation (2.10). Pour le peigne, un vecteur de $nbpts$ éléments est d'abord initialisé à l'aide de zéros. Les positions exactes des deltas de Dirac sont ensuite calculées selon l'équation (2.3). Pour chaque impulsion du peigne, l'élément du vecteur de support dont la valeur est la plus proche est changé à 1.

Ceci a pour effet d'allouer un décalage des noyaux par rapport à leur position théorique de $\pm \Delta t/2$. Aussi, dans le cas des noyaux, le premier élément calculé à la valeur de temps nul

$N(t = 0) = 0$ pour tous les noyaux, le début du noyau se trouve ainsi tronqué pour une longueur de Δt, et la fin d'une valeur variant dans $[0, \Delta t]$. Le noyau est donc défini sur un nombre de points relativement limité, particulièrement lorsque sa largeur $2S$ est petite par rapport au t_{max} de l'ensemble des signaux. C'est en fait ce qui a le plus grand effet sur l'approximation, ce que nous allons montrer par certains tests.

7.2 L'influence sur les images

Afin d'apprécier l'effet de $nbpts$, nous avons tenté de le modifier et de produire des images avec des vecteurs de paramètres $\vec{W}, \vec{S}$ et $\vec{\theta}$ préétablis. Le choix naturel est l'ensemble des paramètres déterminés auparavant comme produisant les lettres améliorées. En théorie, reprendre la production de ces lettres avec une précision parfaite, soit une infinité de points, et obtenir exactement les mêmes images serait une forte confirmation de la validité de l'approximation que nous utilisons. Il est malheureusement impossible d'obtenir cette confirmation, mais nous pouvons tout de même nous en rapprocher en augmentant significativement la valeur de $nbpts$ par rapport à la valeur standard utilisée de 1000. Nous pouvons vouloir approcher l'infini avec une valeur arbitrairement grande, mais encore une fois une limite s'impose. En fait, l'algorithme de la production des signaux par l'approximation du quantron est d'un ordre de complexité $O(n^2)$ en fonction de la précision $nbpts$. En effet, doubler celui-ci double à la fois le nombre d'éléments du support du peigne de Dirac et du support du noyau, quadruplant ainsi le nombre d'opérations effectuées lors de la convolution entre ces derniers. En conséquence, bien que la production d'une image n'a requise que quelques secondes avec une précision de 1000, la même image requiert quelques minutes pour $nbpts = 10000$ et plusieurs heures pour $nbpts = 100000$.

Compte tenu de la quantité de lettres produites et à tester, nous les avons d'abord recréées avec une nouvelle précision significativement plus élevée mais abordable de 10000 points. Dans la grande majorité des cas, nous obtenons exactement les mêmes images qu'avec la précision préalable de 1000 points. Il y a cependant apparition de deux pixels actifs sur l'image du L qui

n'y étaient pas au départ, indiquant une faible influence du changement. Plus significatif est le changement observé sur la lettre N.

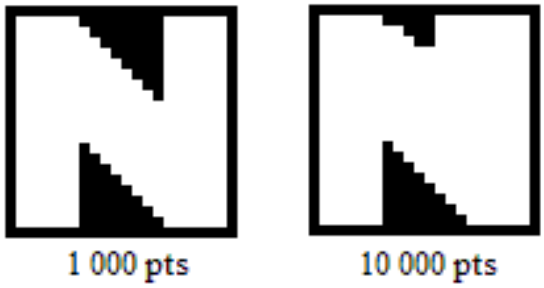

Figure 7.1 : Comparaison entre la lettre N produite avec 1000 points et 10000 points

Nous observons l'épaississement de la bande de droite dans le haut de l'image et une déformation de la diagonale restante. Nous nous sommes donc attardés sur cette lettre afin de découvrir pourquoi elle est si affectée par le changement de précision. Nous avons d'abord effectué l'exercice inverse, soit de diminuer la précision. Il s'avère que l'image perd toute ressemblance à la lettre N lorsque $nbpts$ diminue d'environ 15%. À l'opposé, nous avons alloué à l'algorithme les heures requises pour produire l'image avec 100000 points. Nous avons aussi produit une gamme d'images avec une quantité de points variant entre 1000 et 10000.

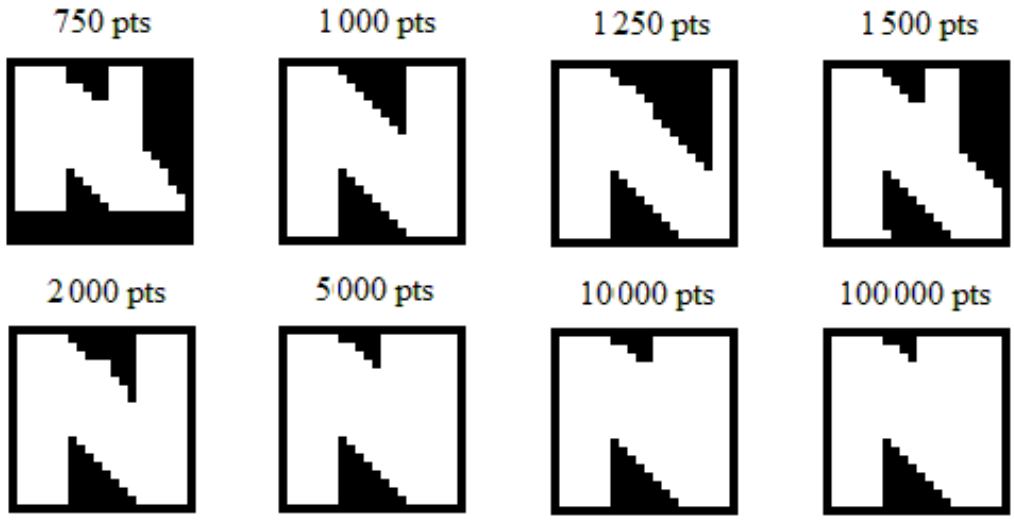

Figure 7.2 : La lettre N produite avec différentes précisions d'approximation

Nous voyons bien la déformation qui résulte d'une baisse de précision à 750 points. Nous voyons aussi que l'image se trouve déjà modifiée avec une simple augmentation de précision de 25%. L'image continue ensuite à se modifier graduellement, une différence faible mais notable se trouvant même lors du passage de 10 000 à 10 0000 points. Notons que l'image est obtenue par une relation logique «*ET*» entre deux quantrons qui contrôlent chacun un des triangles. Individuellement, le triangle du bas subit une légère modification en l'apparition du pixel pour une précision de 1500 points. C'est plutôt l'image qui contrôle le triangle du haut qui est fortement affecté par la précision.

7.3 L'analyse des signaux

Afin de trouver la raison de cet effet, nous avons sélectionné le pixel de coordonnées $(x, y) = (2/20\,, 11/20)$ qui fait partie du triangle du haut. Celui-ci est inactif pour les précisions 1000 à 10000 points mais devient actif pour 100000 points. Nous avons produit les graphiques des signaux dans les trois cas. Voici ce que l'on obtient.

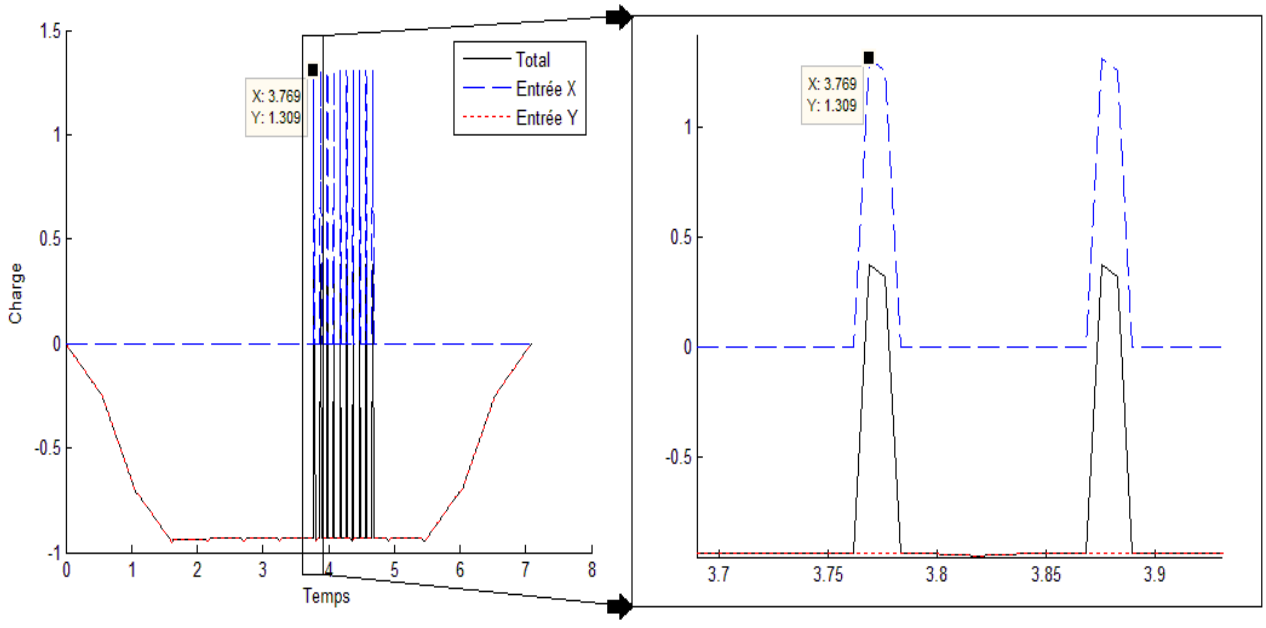

Figure 7.3 : Graphiques des signaux au pixel sélectionné pour 1000 points

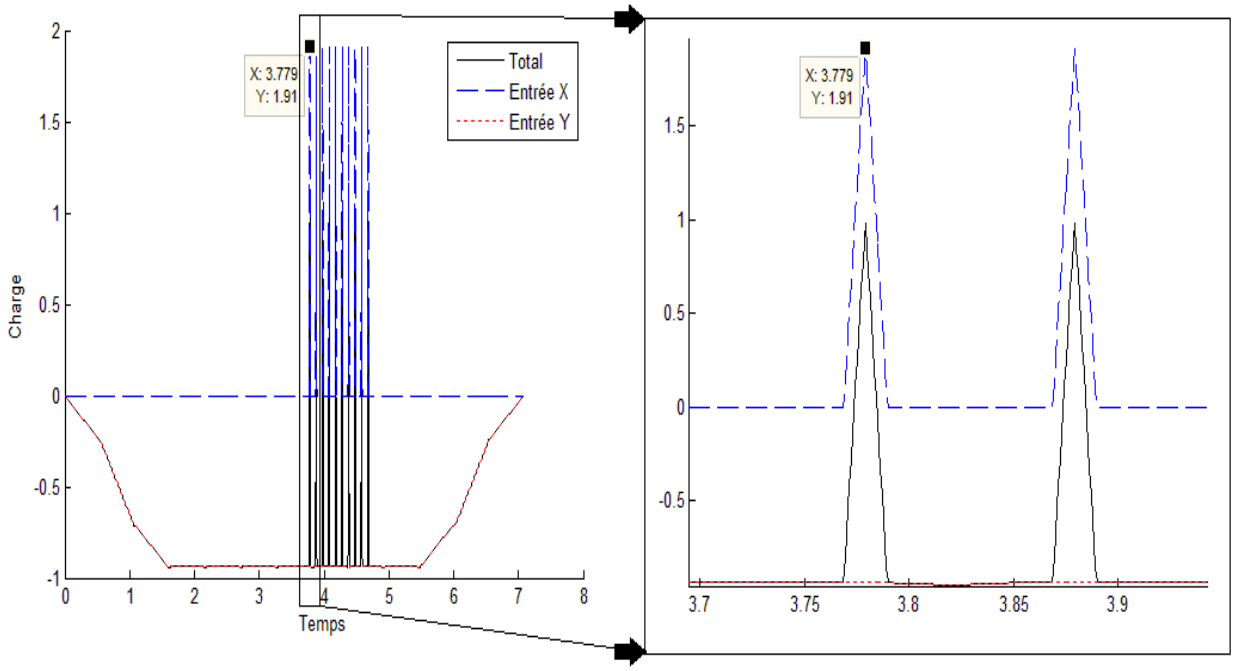

Figure 7.4 : Graphiques des signaux au pixel sélectionné pour 10000 points

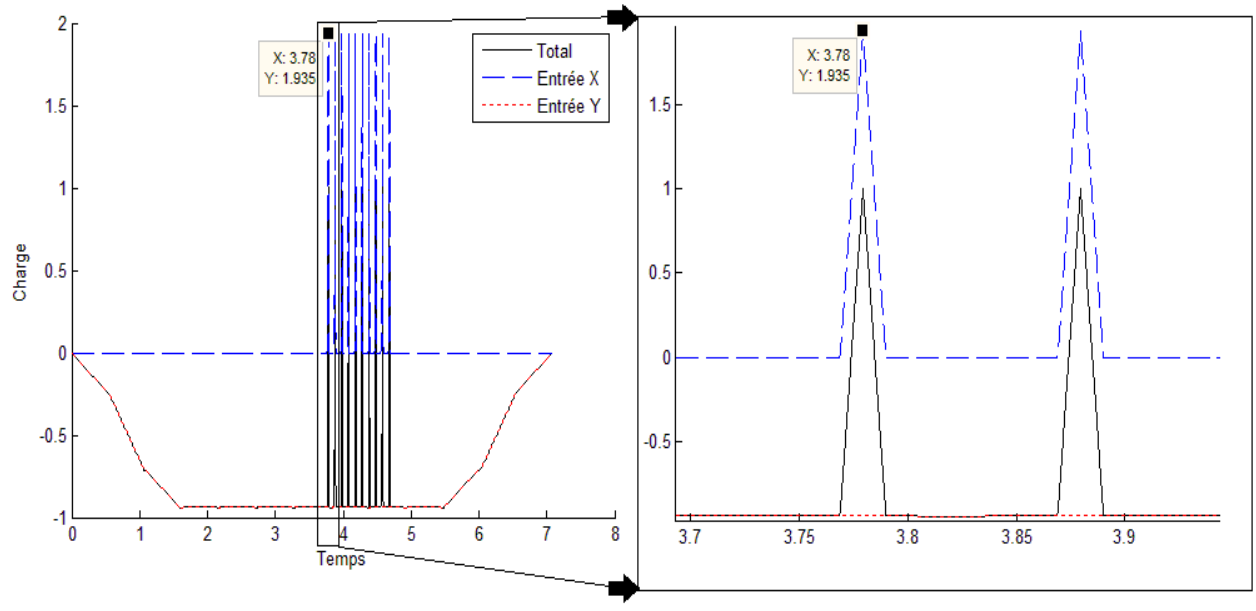

Figure 7.5 : Graphiques des signaux au pixel sélectionné pour 100000 points

Nous remarquons que les noyaux du signal positif sont très étroits par rapport à ceux du signal négatif, soit $S_{x2} = 0{,}0105$ contre $S_{y2} = 1{,}0706$. Notons aussi que leurs hauteurs, soient les valeurs les plus grandes des éléments des vecteurs qui les forment, sont de 1,309, 1,910 et 1,935 respectivement pour les précisions 1000, 10000 et 10000 points, alors que la hauteur théorique est de $w_{x2} = 1{,}9358$.

Pour expliquer cette différence, nous devons revenir aux équations (7.1) et (7.2). Nous avons pour ce pixel :

$$t_{max} = max\{\theta_k + (N-1)X_k + 2S_k\}$$

$$= max\{\theta_{x2} + (10-1)x_2 + 2S_{x2}, \quad \theta_{y2} + (10-1)y_2 + 2S_{y2}\}$$

$$= max\{3{,}7691 + 9 \times 2/20 + 2 \times 0{,}0105, \quad 0 + 9 \times 11/20 + 2 \times 1{,}0706\}$$

$$= max\{4{,}6901, \quad 7{,}0912\} = 7{,}0912$$

$$\Delta t = \frac{t_{max}}{(nbpts - 1)}$$

$$= \frac{7{,}0912}{(1000 - 1)}$$

$$\cong 0{,}0071 \text{ pour } nbpts = 1000.$$

Dans ce cas, le noyau triangulaire est défini sur un vecteur support formé d'un nombre n d'éléments, excluant le premier toujours nul, selon :

$$n = \left\lfloor \frac{2S_{x2}}{\Delta t} \right\rfloor \tag{7.3}$$

$$\cong \left\lfloor \frac{2 \times 0{,}0105}{0{,}0071} \right\rfloor = 2$$

où $\lfloor x \rfloor$ est le plus grand entier inférieur à x. Le triangle étant défini uniquement sur un vecteur de deux éléments, celui-ci est échantillonné à seulement deux endroits ce qui n'est pas suffisant pour obtenir un sommet à la même valeur que son sommet réel à $w_{x2} = 1.9358$. Nous obtenons en fait les valeurs selon les équations du noyau triangle (2.10) pour les deux points.

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} w_{x2}\varphi_T^*(\Delta t) \\ w_{x2}\varphi_T^*(2\Delta t) \end{pmatrix}$$

$$= \begin{pmatrix} w_{x2}\frac{1}{S_{x2}}\{\Delta t u(\Delta t) + [2S_{x2} - 2\Delta t]u(\Delta t - S_{x2}) + [\Delta t - 2S_{x2}]u(\Delta t - 2S_{x2})\} \\ w_{x2}\frac{1}{S_{x2}}\{2\Delta t u(2\Delta t) + [2S_{x2} - 4\Delta t]u(2\Delta t - S_{x2}) + [2\Delta t - 2S_{x2}]u(2\Delta t - 2S_{x2})\} \end{pmatrix} \quad (7.4)$$

$$\cong \begin{pmatrix} 1{,}9358 \times \frac{1}{0{,}0105}\{0{,}0071 \times u(0{,}0071) + [0{,}0068] \times u(-0{,}0034) + [-0{,}0139] \times u(-0{,}0139)\} \\ 1{,}9358 \times \frac{1}{0{,}0105}\{0{,}0142 \times u(0{,}0142) + [-0{,}0074] \times u(0{,}0037) + [-0{,}0068] \times u(-0{,}0068)\} \end{pmatrix}$$

$$\cong \begin{pmatrix} 1{,}3090 \\ 1{,}2537 \end{pmatrix}$$

Le sommet est alors au point central et est environ à 1.3 plutôt qu'à 1.9 comme désigné par le modèle théorique. Alors qu'on augmente la précision par un facteur de dix puis de cent pour obtenir les deux autres images, le nombre d'éléments du vecteur support du noyau, incluant cette fois le premier toujours nul, passe de trois à trente puis à trois cents. Le sommet se rapproche alors de plus en plus du sommet théorique et c'est ce qui fait la différence dans l'image.

Nous en concluons que l'approximation utilisée dans ce travail est particulièrement sensible à la valeur de précision $nbpts$ lorsque qu'un élément du paramètre de largeur de noyau $\vec{S}$ est petit. Nous pouvons établir une certaine borne approximative à partir de laquelle la précision a un effet significatif. Nous avons :

$$t_{max} = max\{\theta_k + (N-1)X_k + 2S_k\}$$

$$t_{max} \cong \{\theta_{max} + (N-1) \times 1\} \quad (7.5)$$

pour l'ensemble de l'image, θ_{max} étant le plus grand des délais. Alors :

$$\Delta t = \frac{t_{max}}{(nbpts - 1)}$$

$$\cong \frac{\{\theta_{max} + (N-1)\}}{(nbpts)} \quad (7.6)$$

considérant $nbpts$ suffisamment grand. Alors, le nombre d'éléments du support du noyau à la plus petite largeur S_{min} sera :

$$n = \left\lfloor \frac{2S_{min}}{\Delta t} \right\rfloor$$

$$\cong \frac{2S_{min}nbpts}{\{\theta_{max}+(N-1)\}}. \tag{7.7}$$

On peut établir approximativement la valeur de S_{min} pour obtenir un certain nombre de points n au support du noyau.

$$S_{min} \geq \frac{n\{\theta_{max}+(N-1)\}}{2nbpts} \tag{7.8}$$

Considérons maintenant le nombre d'éléments n requis pour éviter la variation du sommet. Les n points sont répartis sur la longueur $2S$ du noyau. Il y aura, pour un n pair, $n/2$ points sur la pente ascendante du triangle et $n/2$ points sur la pente descendante. Pour un n impair, il y aura $(n+1)/2$ points sur une des pentes et $(n-1)/2$ points sur l'autre. Il y a donc toujours au moins $n/2$ points sur une des pentes. Le point le plus haut sur cette pente sera alors toujours, au minimum, à la fraction $(n/2)/(n/2+1)$. Donc si nous désirons que le sommet des noyaux atteigne au minimum un pourcentage ω de la hauteur réelle, nous avons :

$$\omega \geq 100\frac{(n/2)}{(n/2+1)} \tag{7.9}$$

$$n \geq \frac{2\omega/100}{(1-\omega/100)}. \tag{7.10}$$

En combinant les équations (7.8) et (7.10),on obtient :

$$S_{min} \geq \frac{\omega\{\theta_{max}+(N-1)\}}{(100-\omega)nbpts}. \tag{7.11}$$

Cette relation impose une certaine limite inférieure aux valeurs de $\vec{S}$ en fonction des paramètres de délais $\vec{\theta}$, du nombre de noyau N et de la précision $nbpts$ afin d'assurer que partout dans l'image, les sommets des noyaux de forme triangulaire puissent atteindre au minimum le pourcentage ω de leur valeur réelle. Nous pouvons par exemple déterminer le S_{min} pour le cas qui nous intéresse ici, si l'on fixe le pourcentage à atteindre à 90% pour 1000 points.

$$S_{min} \geq \frac{\omega\{\theta_{max} + (N-1)\}}{(100-\omega)nbpts}$$

$$S_{min} \geq \frac{90\{3{,}7691 + (10-1)\}}{(100-90)1000}$$

$$S_{min} \geq 0{,}1149$$

Ce qui n'est clairement pas respecté. Nous pouvons aussi modifier légèrement l'inégalité afin d'obtenir la précision nécessaire $nbpts$ afin d'obtenir le pourcentage ω en fonction des paramètres fixés. Dans ce cas, toujours pour $\omega = 90$,

$$nbpts \geq \frac{\omega\{\theta_{max}+(N-1)\}}{(100-\omega)S_{min}} \tag{7.12}$$

$$nbpts \geq \frac{90\{3{,}7691 + (10-1)\}}{(100-90)0105}$$

$$nbpts \geq 10945.$$

Nous pouvons vérifier cette équation pour toutes les lettres présentement produites.

Tableau 7-1 : Nombres de points nécessaires pour l'obtention de différents pourcentages de hauteurs de sommets pour l'ensemble des lettres produites

Lettre produite		θ max	S min	nbpts min		
	# du quantron			90%	95%	99%
A	1	1,5071	0,1266	747	1577	8216
	2	6,4837	1,4500	96	203	1057
	3	1,8022	0,3333	292	616	3209
C	1	7,6765	0,4069	369	779	4057
E	1	7,6146	0,6707	223	471	2452
	2	5,957	0,2278	591	1248	6500
F	1	4,9413	0,2782	451	952	4961
H	2	7,1942	0,4082	357	754	3928
	2	7,1942	0,4082	357	754	3928
I	1	6,1225	0,3348	407	858	4472
J	1	5,0264	1,6346	77	163	850
L	1	6,1019	0,0155	8769	18512	96457
M	1	7,6146	0,6707	223	471	2452

	2	5,957	0,2278	591	1248	6500
N	1	3,349	0,3464	321	677	3529
	2	3,7691	0,0105	10945	23106	120394
O	1	27,1942	0,4082	798	1685	8778
P	1	9,6786	0,3484	483	1019	5308
T	1	10,4986	0,8784	200	422	2198
	2	4,7448	0,3557	348	734	3826
U	1	7,1942	0,4082	357	754	3928
	1	0	0,7200	113	238	1238

Les précisions nécessaires sont calculées en nombre de points pour obtenir des sommets à différents pourcentages selon les paramètres des quantrons qui produisent chaque image. Les valeurs qui dépassent les 1000 points utilisés sont surlignées en gris. Nous voyons que la majorité des lettres répondent au critère de 90%. Seules les lettres N et L requièrent un nombre plus grand de points. Ce sont d'ailleurs les seules où nous avons observé un effet de l'augmentation de la précision à 10000 points. Les nombres de points requis sont d'ailleurs fortement plus élevés que pour les autres lettres. En augmentant le critère à 95% et 99%, l'on perd d'abord la moitié des lettres puis la totalité. Il en demeure pour 95% que le nombre de points ne dépasse pas 2000 pour les lettres autres que les deux qui posaient déjà problème. Nous croyons que le critère 90% est suffisant pour détecter les cas problèmes, mais comme nous le montrons dans un instant, il faut tout de même être prudent.

Nous avons noté une autre différence faisant apparition dans l'image de la lettre N lorsque la précision est modifiée, un pixel unique apparaît au bas de l'image pour $nbpts = 1500$. Pourtant, selon les équations établies, les largeurs des noyaux pour le quantron responsable de cette partie de l'image sont suffisamment grandes pour que les sommets ne varient pas fortement. En fait, nous pouvons approximer cette variation maximale avec :

$$\omega \leq \frac{nbpts \times S_{min}}{\{\theta_{max} + (N-1)\} + nbpts \times S_{min}} \tag{7.13}$$

$$\omega \leq \frac{1000 \times 0{,}3464}{\{3{,}3490 + (10-1)\} + 1000 \times 0{,}3464}$$

$$\omega \leq 0{,}9657.$$

Les sommets ne sont donc jamais inférieurs à 96% de leurs valeurs réelles. Voyons les graphiques des signaux pour le pixel en question aux précisions 1000 et 1500 points.

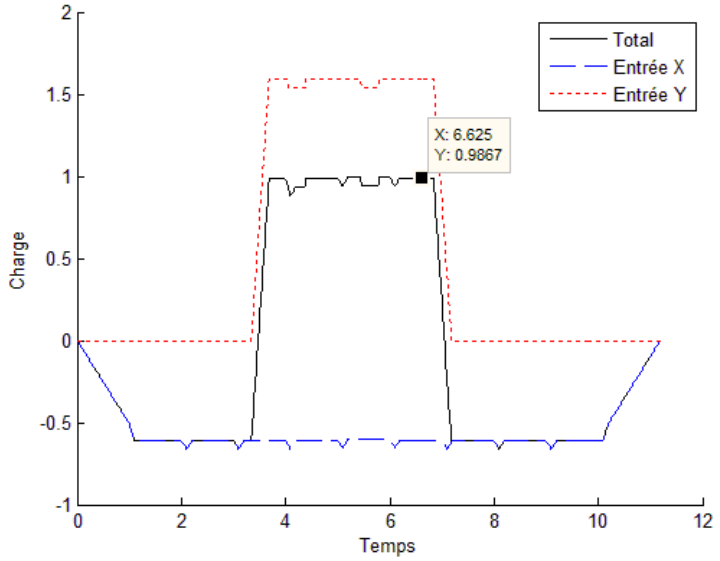

Figure 7.6 : Graphique des signaux au pixel d'intérêt pour 1000 points

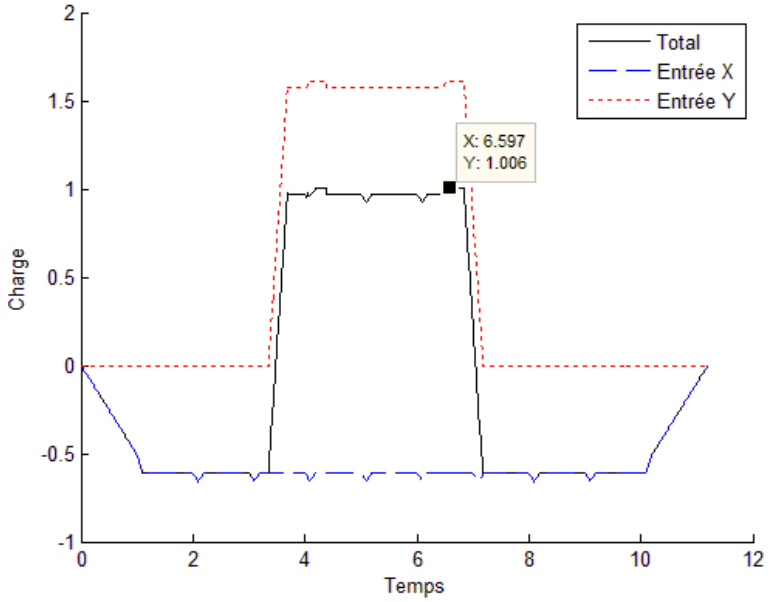

Figure 7.7 : Graphique des signaux au pixel d'intérêt pour 1500 points

Les maximas des signaux additionnés se trouvent à 0,9867 et 1,006. Nous nous trouvons ici sur un cas où le maximum du signal total est suffisamment près de la valeur du seuil pour qu'une variation d'à peine 2% affecte la sortie. Ceci nous avertit que même en respectant des conditions plutôt strictes sur le pourcentage des sommets ω, il est possible que des défauts persistent dans des cas limites.

7.4 Conclusions sur la précision de l'approximation

L'objectif était ici de justifier la validité de l'approximation numérique utilisée pour calculer les signaux internes du quantron. Idéalement, nous aurions voulu comparer les résultats de l'approximation aux résultats théoriques, mais ceci s'avère impossible parce que définis sur un domaine continu. Par contre, en comparant les résultats obtenus pour la précision utilisée au cours du travail à des précisions plus grandes, nous avons montré que la majorité des résultats restent valides. Une exception nous a menés à analyser plus en profondeur les signaux et à observer la conséquence néfaste possible d'une résolution trop faible. Nous en avons déduit des règles qui permettent de calculer la précision requise en fonction des paramètres du quantron simulé. Ces règles peuvent servir soit à choisir d'avance la précision de l'approximation à utiliser pour un ensemble de paramètres donné, ou à valider un ensemble de résultats en vérifiant si la précision est suffisante pour chaque cas. Par exemple, le tableau 7-1 nous permet immédiatement de détecter une insuffisance pour la lettre N. Les équations trouvées sont en conséquence du noyau « Triangle » choisi pour obtenir les images de ce travail. Il serait possible de modifier l'équation (7.12) afin d'adapter les résultats aux noyaux « Vague » et « Parabole ». Dans le cas du noyau « Carré », l'influence de la précision se trouve plutôt dans les troncatures de ses extrémités comme nous l'avons noté dans la sous-section traitant du problème XOR. La prochaine section s'attarde sur une seconde approximation imposée au modèle, la limite du nombre de noyaux.

8 NOMBRE DE NOYAUX

Nous avons expliqué plus tôt les raisons pour lesquelles il est nécessaire de limiter le nombre de noyaux disponibles à la construction des signaux du quantron. Ayant déjà une grande quantité de paramètres à faire varier pour explorer les images pouvant être produites par le modèle, nous avons décidé de fixer le nombre de noyaux à une valeur que nous avons choisie $N = 10$. Cette valeur est estimée comme suffisamment grande pour permettre aux signaux de se construire et allouer des interactions intéressantes tout en étant suffisamment petite pour ne pas encombrer les calculs inutilement. L'hypothèse est que nous arrivons à observer la majeure partie du potentiel de création d'images du quantron avec cette valeur de N. Autrement dit, ni l'augmenter, ni la diminuer, ne permettrait des interactions inédites Afin de justifier notre choix, nous étudions ici les effets de la modification du nombre de noyaux N sur la production de quelques lettres aux paramètres connus.

8.1 Exemples d'analyses des effets de N sur des lettres

8.1.1 Effets sur la lettre C

Nous commençons par étudier l'effet d'une modification du nombre de noyaux sur une lettre simple dont l'interaction entre les signaux X et Y nécessaire pour la produire est bien comprise. La lettre choisie est le C. Nous n'avons pas effectué d'étude des signaux spécifiquement pour cette lettre, celle-ci ayant déjà la forme désirée, mais notons que nous avons analysé les signaux qui la produisent lorsque nous avons étudié la lettre E. Il s'agit de deux bandes, horizontale en haut et verticale à gauche de l'image, produites indépendamment par les signaux X et Y, puis une seconde bande horizontale au bas de l'image produite cette fois lorsque le signal X atteint le début du signal Y. Voici la lettre originale avec dix noyaux.

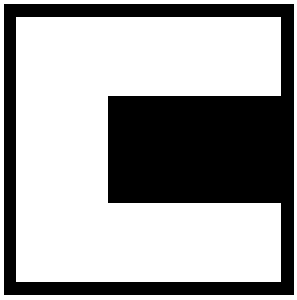

Figure 8.1 : Image de la lettre C originale avec $N = 10$ noyaux

Afin de visualiser l'effet du changement du nombre de noyaux, nous avons reproduit les images avec exactement les mêmes paramètres, mais avec des valeurs de $N \in \{1,2,\dots,15\}$, voici la transition obtenue.

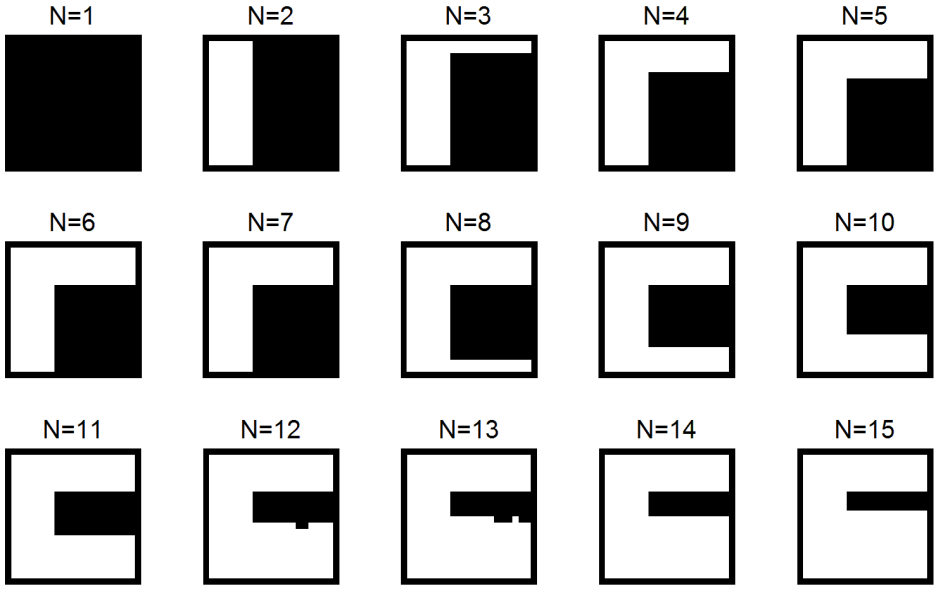

Figure 8.2 : Images de la lettre C pour un nombre de noyaux de 1 à 15

Regardons d'abord l'effet du nombre de noyaux sur les bandes indépendantes en haut et à gauche de l'image. La bande de gauche est conservée intacte alors que le nombre de noyaux augmente ou diminue et ce de $N = 2$ à 15. Elle n'est perdue que lorsqu'il ne reste qu'un noyau. Les

paramètres régissant le signal sont $w_y = 0{,}9468\ et\ S_y = 0{,}4069$. La hauteur des noyaux est suffisamment proche du seuil pour que celui-ci soit atteint avec la contribution d'un seul autre noyau voisin, c'est pourquoi la bande est conservée pour un nombre de noyaux aussi bas que deux mais est totalement perdue pour $N = 1$. C'est ensuite le paramètre de largeur S_y qui régit à partir de quelle valeur d'entrée la superposition se fera et donc l'épaisseur de la bande dans l'image. Pour une demi-largeur donnée, nous savons que le début du k^e noyau arrivera à contribuer au sommet du premier noyau pour une valeur d'entrée x donnée par :

$$x \leq \frac{S_x}{k-1} \text{ pour } k > 1. \tag{8.1}$$

Alors, selon la convention des valeurs d'entrées en fonction des positions des pixels dans l'image, nous avons :

$$p \leq \frac{S_x \times L}{k-1} \text{ pour } k > 1. \tag{8.2}$$

Avec $S_y \cong 0{,}4$ c'est à partir du pixel 8 ($p \leq 0{,}4 \times 20$) que le début du second noyau arrive à contribuer au maximum du premier. Dans ce cas, c'est à partir du 7^e pixel que la contribution est suffisante pour atteindre le seuil. Augmenter le nombre de noyaux n'aide pas à accélérer cette rencontre et n'affecte donc pas la largeur de la bande.

Quant à elle, la bande du haut reste intacte pour $N = 5$ à 15. Elle s'amincit pour $N = 3$ et 4 puis disparait entièrement pour $N \leq 2$. Les paramètres régissant le signal sont $w_x = 0{,}3644\ et\ S_x = 0{,}8141$. La hauteur des noyaux est significativement plus basse que pour le cas précédent, mais leurs plus grandes largeurs compensent pour obtenir la même bande lorsque $N \geq 5$. La hauteur se situe entre la demie et le tiers du seuil, ce qui explique pourquoi il est impossible d'obtenir l'activation avec seulement deux noyaux ou moins. Il faut en effet au minimum la contribution de trois noyaux pour atteindre le seuil.

Selon la demi-largeur $S_x \cong 0{,}8$, le troisième noyau contribue pour, $p \leq (0{,}8 \times 20)/2 \leq 8$. Cette contribution n'est d'abord pas suffisante et ce n'est qu'à partir de $p = 2$ que l'activation est possible, ce, lorsque le nombre de noyaux est limité à 3. Lorsque $N = 4$, le quatrième noyau arrive à contribuer pour $p \leq (0{,}8 \times 20)/3 \leq 5, \overline{3}$. Nous observons effectivement l'activation à

partir du pixel 5, la somme des trois noyaux précédents est alors déjà suffisamment forte pour que la faible contribution du quatrième parvienne à atteindre le seuil. Ensuite, pour $N = 5$, $p \leq (0{,}8 \times 20)/4 \leq 4$. Le cinquième noyau n'arrive à contribuer au premier que pour les pixels quatre et moins. Or les quatre premiers noyaux arrivent déjà à activer la sortie pour les pixels 5 et moins. Malgré cela, nous voyons apparaitre l'activation pour le pixel 6 lorsque le nombre de noyaux passe à 5.

C'est qu'en vérité, le maximum global du signal constitué de N noyaux se situe au centre de celui-ci, et non au centre du premier noyau. Le dernier noyau peut alors contribuer au maximum si son début précède le centre du signal selon la condition :

$$x(N-1) \leq \frac{x(N-1)+2S_x}{2}, \tag{8.3}$$

donc pour une valeur d'entrée donnée par :

$$x \leq \frac{2S_x}{N-1} \text{ pour } N \geq 2 \tag{8.4}$$

ou bien à partir du pixel :

$$p \leq \frac{2S_x L}{N-1} \text{ pour } N \geq 2. \tag{8.5}$$

Alors, dans le cas du C avec $N = 5$, $p \leq (2 \times 0{,}8 \times 20)/4 \leq 8$, ce qui permet l'activation du pixel 6. Par contre, pour $N = 6$, $p \leq (2 \times 0{,}8 \times 20)/4 \leq 6{,}4$. Le sixième noyau n'arrive à contribuer au maximum du signal qu'à partir du sixième pixel. Or les cinq premiers noyaux arrivent déjà à activer la sortie à ce moment. L'ajout de noyaux supplémentaires n'a alors plus d'effet sur la bande présente dans l'image.

Attardons-nous maintenant à la bande horizontale présente au bas de l'image. À partir de sa largeur originale lorsque $N = 10$, celle-ci s'amincit lorsque le nombre de noyaux diminue jusqu'à disparaitre entièrement pour $N = 7$. Au contraire, elle prend de l'ampleur alors que le nombre de noyaux augmente. Ceci n'est pas une surprise puisque $w_y = 0{,}9468$ est très près du seuil. Alors que le signal Y n'arrive plus à se superposer pour activer la sortie par lui-même, il n'a besoin que de la contribution d'un seul noyau du signal X pour atteindre le seuil. Or l'étendue du

signal X est directement proportionnelle au nombre de noyaux qui le compose. Nous pouvons voir la différence entre les situations à $N = 10$ et $N = 5$ pour un point de la bande à l'aide des graphiques suivants.

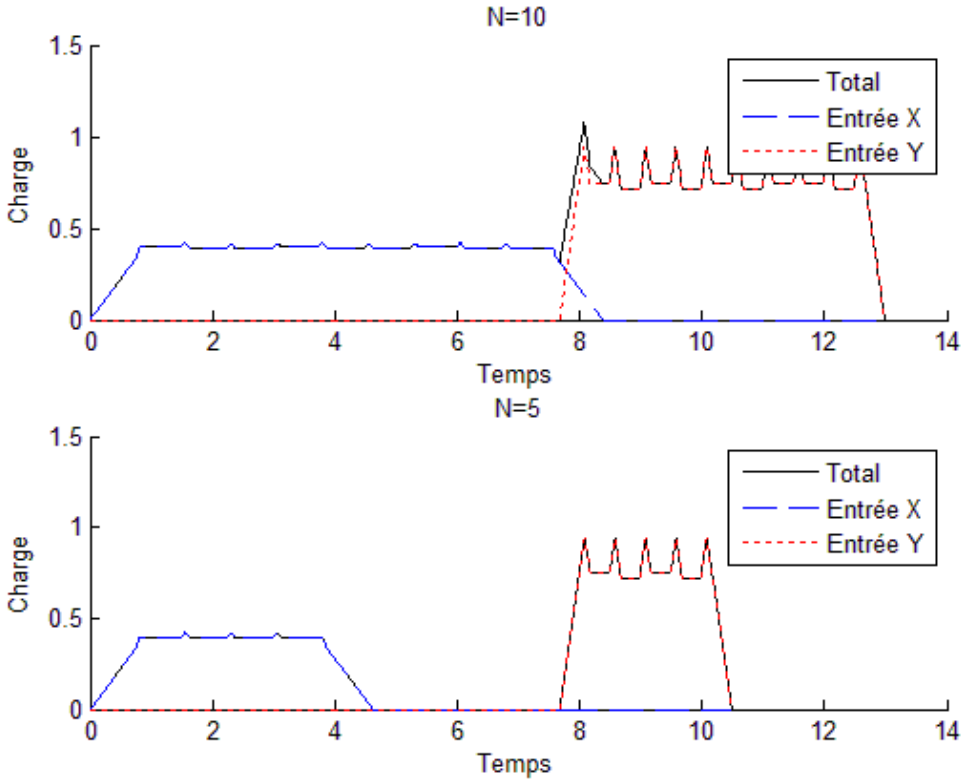

Figure 8.3 : Graphiques des signaux du pixel $(15,10)$ pour la lettre C avec 10 et 5 noyaux

Nous avons déjà vu comment les deux bandes « indépendantes » restent inchangées pour un nombre de noyaux diminué jusqu'à $N = 5$, ou bien augmenté indéfiniment. Bien que l'on voit disparaître ou allonger la bande « interactive », celle-ci est le produit d'une interaction suffisamment simple pour que l'on puisse contrer le changement à l'aide d'un seul paramètre. Considérant cette bande comme le produit de la simple atteinte du début du signal Y par la fin du signal X, nous n'avons qu'à diminuer ou augmenter la valeur de θ_y en conséquence du retrait ou de l'ajout de noyaux. Ainsi, pour arriver à atteindre le signal Y avec la moitié du nombre de

noyaux, il a suffi de réduire θ_y à 4. Nous obtenons alors, pour le même pixel que les graphiques précédents, les signaux suivants.

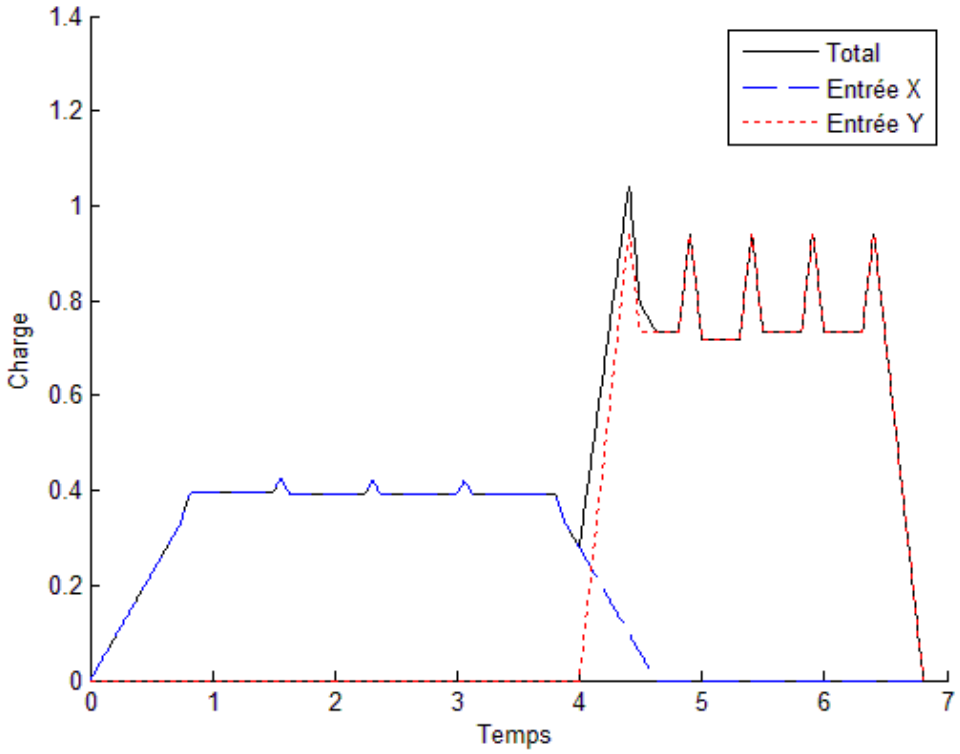

Figure 8.4 : Graphique des signaux du pixel (15,10) de la lettre C lorsque $N = 10$ et $\theta_y = 4$

L'image de la lettre C est alors identique à l'originale. Dans le cas d'une image simple telle celle de la lettre C, on voit qu'il est possible d'obtenir les mêmes résultats avec moins de noyaux. À la limite, seulement deux noyaux pourraient ici produire le même résultat. Ceci n'est pas le cas pour toutes les images possibles, la prochaine lettre en est un exemple.

8.1.2 Effets sur la lettre *J*

Nous pouvons répéter l'expérience cette fois avec une lettre dont la partie « interactive » est plus complexe, soit la lettre J, avec encore une fois des valeurs de $N \in \{1,2,\ldots,15\}$. Voici la transition obtenue.

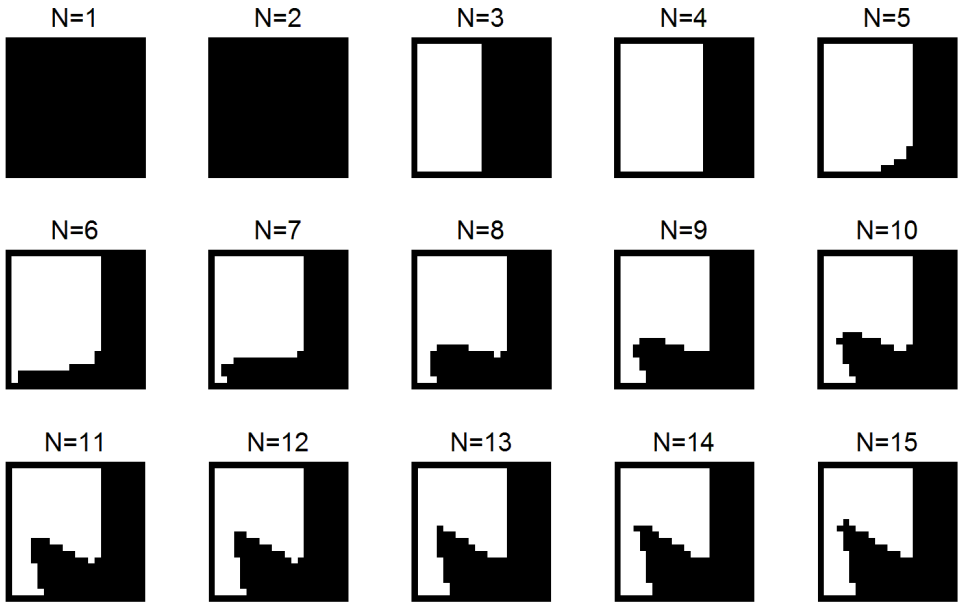

Figure 8.5 : Images de la lettre J pour un nombre de noyaux de 1 à 15

Notons que pour la lettre J, nous inversons normalement les pixels. Nous ne l'avons pas fait pour cette analyse afin de la simplifier visuellement. Rappelons les valeurs des paramètres pour ces images.

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} -0{,}6659 \\ 0{,}4188 \\ 2{,}4496 \\ 1{,}6346 \\ 0 \\ 5{,}0264 \end{pmatrix}. \tag{8.6}$$

Le signal Y est le seul positif, il n'y a donc pas de bande blanche horizontale. Nous voyons la bande verticale apparaître à partir de $N = 3$ puis se stabiliser à partir de $N = 5$. Considérant la valeur de $1/3 \leq w_y \leq 1/2$, la disparition de la bande entre $N = 1$ et $N = 2$ n'est pas une surprise. La largeur de bande maximale est observée constante à $p = 14$ pour $N \geq 5$. En effet, pour $N \geq 6$ nous avons :

$$p \leq \frac{2 \times 1{,}6346 \times 20}{6 - 1} \leq 13{,}0768.$$

Les noyaux supplémentaires n'arrivent donc pas à augmenter le maximum pour les pixels plus grands que 13.

Nous observons aussi l'absence d'influence de l'entrée X pour $N \leq 4$. Nous avons une valeur de délai nulle pour le signal X et $\theta_y = 5{,}0264$. Pour que le signal négatif X ait une influence sur l'image, il doit arriver à atténuer le maximum du signal positif pour une valeur d'entrée Y qui lui permet d'atteindre le seuil. Pour ce faire, il faut au minimum que la fin du signal négatif atteigne ce maximum situé au centre du signal positif. La condition se traduit ainsi.

$$x(N-1) + 2S_x \geq \theta_y + \frac{y(N-1) + 2S_y}{2}$$

$$N \geq \frac{2\theta_y + 2S_y - 4S_x}{2x - y} + 1 \tag{8.7}$$

Considérant que la bande a une épaisseur minimale de $p = 10$ pixels et donc une activation de la sortie quand $y \leq 0{,}5$ pour $N \geq 3$ et que la valeur maximale de x est 1, nous pouvons trouver la valeur minimale de N pour obtenir une interaction.

$$N \geq \frac{2 \times 5{,}0264 + 2 \times 1{,}6346 - 4 \times 2{,}4496}{2 - 0{,}5} + 1 \geq 3{,}35$$

Ceci est suffisant pour expliquer le manque d'interaction pour $N = 3$. Pour $N = 4$ nous en déduisons que la fin du signal négatif arrive bien à atteindre le sommet positif, mais que son atténuation n'est pas encore suffisamment prononcée.

À partir de $N = 5$, nous voyons tranquillement apparaître la patte du J. Celle-ci commence à prendre une forme courbe à partir de $N = 8$ puis son extrémité prend une forme plus pointue lorsque N continue à augmenter puis semble se stabiliser à partir de $N = 13$. Nous en concluons qu'il n'est effectivement pas nécessaire d'utiliser plus de dix noyaux pour obtenir le type d'interaction qui produit l'image de la lettre J. Nous avons déjà vu qu'une quantité trop petite de noyaux déforme la lettre, mais est-il tout de même possible d'obtenir une image semblable en modifiant les paramètres? C'est ce que nous avons tenté de faire avec $N = 5$, la moitié du nombre de noyaux habituel.

Pour arriver aux nouveaux paramètres, nous avons utilisé la même démarche que dans la section « étude des signaux ». Par contre, il a d'abord été nécessaire d'effectuer certains ajustements de forte envergure pour deux des paramètres. Pour arriver à choisir ces ajustements, nous avons comparé les signaux obtenus avec cinq noyaux à ceux de dix noyaux pour trois pixels importants sélectionnés en fonction des particularités qui caractérisent la lettre. Le premier pixel $(14,4)$ se situe au sommet de la pointe du J, le second $(14,12)$ à sa droite et le troisième $(19,4)$ en dessous du premier. De cette façon, pour l'image originale non inversée à 10 noyaux, le pixel 1 est inactif, puis passer de celui-ci à un des autres crée l'activation. Voici les nouveaux signaux pour $N = 5$ adjacents à leurs équivalents pour $N = 10$.

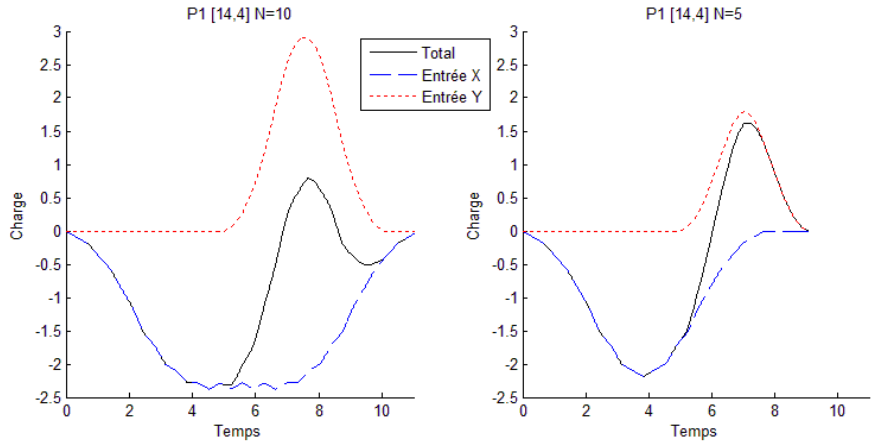

Figure 8.6 : Signaux au pixel 1 $(14,4)$ le la lettre J pour 10 et 5 noyaux

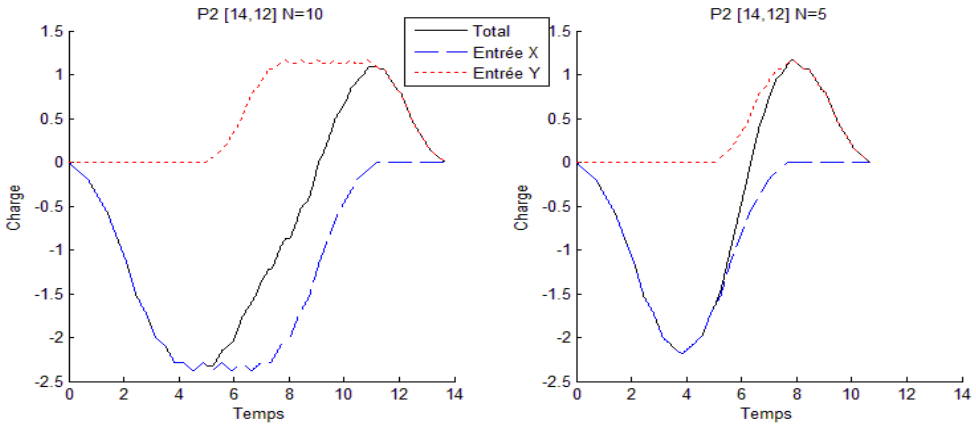

Figure 8.7 : Signaux au pixel 2 $(14,12)$ le la lettre J pour 10 et 5 noyaux

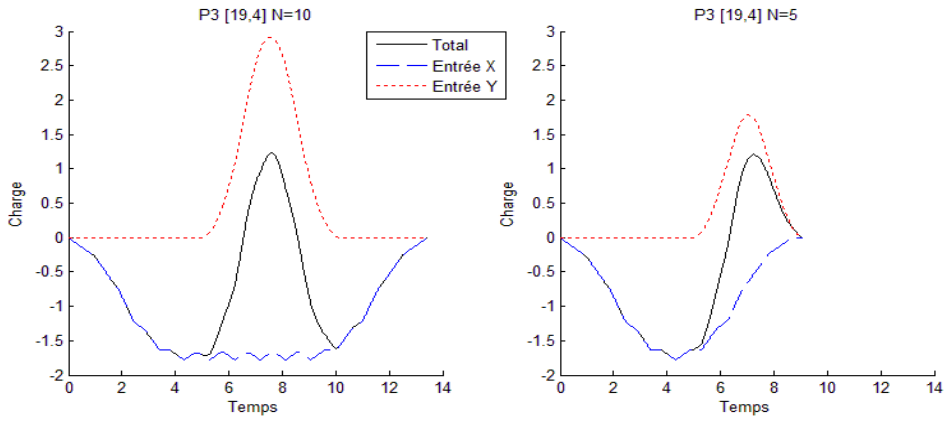

Figure 8.8 : Signaux au point 3 (19,4) le la lettre J pour 10 et 5 noyaux

Dans les trois cas, mais particulièrement pour les pixels 1 et 3, la diminution du nombre de noyaux fait que le signal décalé Y se retrouve trop loin par rapport au signal négatif X pour obtenir les mêmes effets. Cette observation nous a menés à diminuer la valeur du délai θ_y. Ensuite, nous voyons que pour tous les points, la quantité limitée de noyaux fait que le signal négatif obtenu n'a qu'un sommet unique contrairement aux cas où $N = 10$. Pour obtenir des plateaux et corriger cette différence, nous avons diminué la valeur de la largeur des noyaux S_x. Afin d'arriver le plus près possible des graphiques à dix noyaux, nous avons tenté plusieurs nouvelles valeurs pour aboutir à $\theta_{y2} = \theta_y/3$ et $S_{x2} = S_x/2$. Ensuite, par la méthode d'amélioration utilisée lors de l'étude des signaux, nous avons légèrement modifié les poids w_x et w_y afin d'obtenir l'image du J la plus fidèle à l'originale. Les nouveaux paramètres sont :

$$\begin{pmatrix} w_{x2} \\ w_{y2} \\ S_{x2} \\ S_{y2} \\ \theta_{x2} \\ \theta_{y2} \end{pmatrix} = \begin{pmatrix} -0{,}6799 \\ 0{,}4688 \\ 1{,}2248 \\ 1{,}6346 \\ 0 \\ 1{,}6755 \end{pmatrix}. \tag{8.8}$$

Voici, pour les mêmes trois points, les signaux modifiés obtenus avec $N = 5$ toujours comparés à leur équivalent pour $N = 10$.

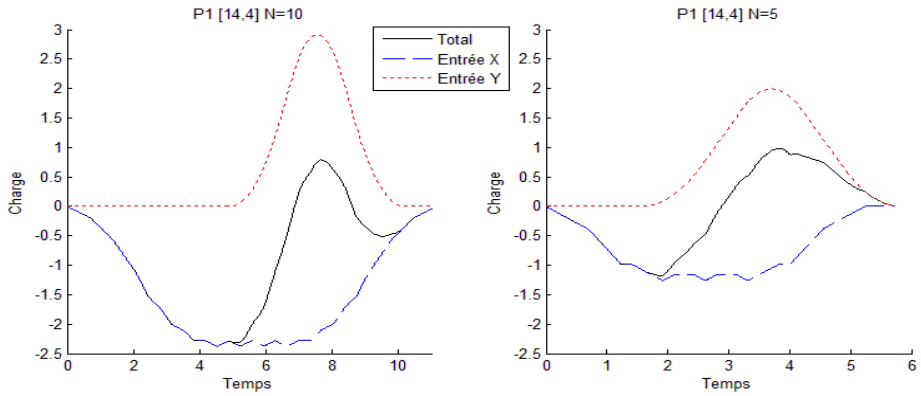

Figure 8.9 : Signaux modifiés au pixel 1 (14,4) de la lettre *J* pour 10 et 5 noyaux

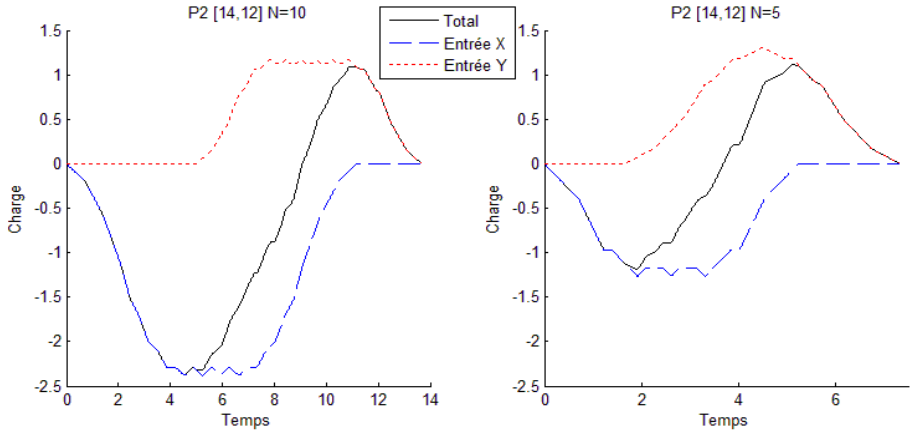

Figure 8.10 : Signaux modifiés au pixel 2 (14,12) de la lettre *J* pour 10 et 5 noyaux

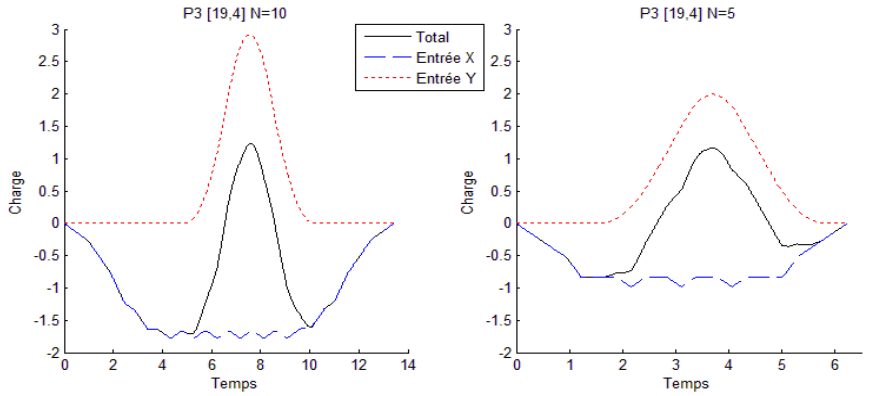

Figure 8.11 : Signaux modifiés au pixel 3 (19,4) de la lettre *J* pour 10 et 5 noyaux

L'on peut y voir que nous arrivons à une interaction similaire entre les signaux, ce qui donne l'image suivante.

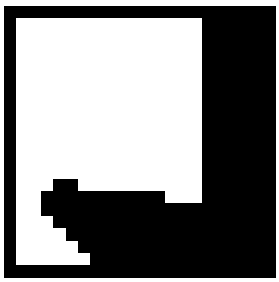

Figure 8.12 : Image de la lettre J pour 5 noyaux avec modification des paramètres

La nouvelle image, bien qu'elle possède tout de même les caractéristiques principales de la lettre J, n'est pas identique à l'originale. Diminuer le nombre de noyau de moitié nous empêche d'obtenir exactement les mêmes résultats qu'avec $N = 10$, contrairement à la lettre C.

8.1.3 Conclusions sur l'influence du nombre de noyaux

Au cours de cet exercice, nous avons tenté de valider notre hypothèse de départ sur le nombre de noyaux nécessaires pour obtenir des images intéressantes et variées. Le choix de $N = 10$ nous permet de construire une quantité d'interactions qu'un plus grand nombre de noyaux n'augmente pas significativement tandis qu'un nombre plus petit rend plus limité. Il n'est toutefois pas impossible que l'emploi de différents nombres de noyaux permette d'obtenir des images inédites avec $N = 10$, mais les contraintes en temps de calcul imposent une limite. Cependant, ces pertes en raffinement des images peuvent être retrouvées dans un MLQ par l'ajout de quantrons supplémentaires. La prochaine section utilise l'expérience acquise afin d'offrir une méthode relativement simple pour créer n'importe quelle surface discriminante désirée.

9 CRÉATION DE RECTANGLES

Nous avons, au cours des sections précédentes, observé et étudié plusieurs signaux, leurs types d'interaction et les images qu'ils produisaient afin d'en déduire les relations qui les lient ensemble. Lors de l'étude des lettres obtenues, ces observations nous ont permis de modifier les paramètres des quantrons en fonction de leurs effets afin d'obtenir des images de lettres plus soignées. Pour les formes nécessitant des interactions plus complexes, telles la courbe de la lettre J ou bien les diagonales de la lettre N, nous sommes bien arrivés à les modifier à notre guise, mais nous n'en tirons pas de règles générales sur les paramètres nécessaires à leur obtention. C'est-à-dire que pour les reproduire, nous devons partir des paramètres originaux et non les produire à partir de certaines règles ou équations. Par contre, pour les formes plus simples, nous décrivons dans cette section les relations que nous avons déduites et qui permettent d'obtenir des résultats intéressants.

9.1 Les bandes indépendantes

Nous observons d'abord que les bandes horizontales et verticales en haut et à gauche des images dépendent uniquement et respectivement des entrées X et Y. Celles-ci ne requièrent aucune interaction entre les signaux. Il est envisageable d'établir une règle simple qui nous permet de déterminer les paramètres w et S qui produisent une bande de largeur désirée B. La plus simple est :

$$w = \Gamma - d \tag{9.1}$$

$$S = \frac{(B+0,5)}{L}, \tag{9.2}$$

Γ étant la valeur du seuil et d est la plus petite unité utilisée pour définir les paramètres. Nous avons toujours utilisé quatre décimales pour définir nos paramètres, ainsi avec le seuil établi à 1, nous avons :

$$w = 1 - 0,0001 = 0,9999.$$

Le but est d'avoir un noyau dont la hauteur est très proche du seuil sans l'atteindre. Il n'a besoin que d'un minimum d'aide du noyau suivant pour activer la sortie. Nous pouvons ainsi contrôler le moment où le début du second noyau atteint le sommet du premier selon l'équation (8.2) où $k = 2$ et où nous remplaçons p par B pour la « largeur de la bande » désirée en pixels. Nous ajoutons aussi un demi-pixel afin de contrer les effets de l'approximation et de nous assurer que le pixel désiré s'active bien. Nous avons par exemple pour une bande de 10 pixels sur les 20 que contient l'image :

$$S = \frac{(10 + 0{,}5)}{20} = 0{,}5250.$$

Les paramètres du vecteur de délais $\vec{\Theta}$ n'ont pas d'influence sur ces bandes. Par contre si nous voulons nous assurer qu'il n'y ait pas d'interaction, nous devons décaler un des signaux suffisamment pour qu'ils ne se rencontrent jamais. Ceci est possible, par exemple pour le signal Y, en fixant le délai selon :

$$\theta_y \geq (N - 1) + 2S_x. \tag{9.3}$$

Nous produisons ici un exemple d'image à partir de ces premières règles. Soit les largeurs de bande $B_x = 10$ et $B_y = 5$, les valeurs des paramètres $w_x \; et \; S_x$ sont celles déterminées plus haut, puis :

$$w_y = 1 - 0{,}0001 = 0{,}9999$$

$$S_y = \frac{(5 + 0{,}5)}{20} = 0{,}2750$$

$$\theta_y = (10 - 1) + 2 \times 0{,}5250 = 10{,}05.$$

Ainsi avec les paramètres :

$$\begin{pmatrix} w_x \\ w_y \\ S_x \\ S_y \\ \theta_x \\ \theta_y \end{pmatrix} = \begin{pmatrix} 0{,}9999 \\ 0{,}9999 \\ 0{,}5250 \\ 0{,}2750 \\ 0 \\ 10{,}0500 \end{pmatrix}, \tag{9.4}$$

nous obtenons l'image suivante.

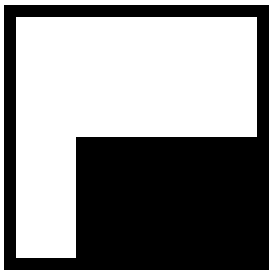

Figure 9.1 : Image de bandes indépendantes de largeurs $B_x = 10$ et $B_y = 5$

Ces équations sont simplifiées au maximum. Nous fixons les valeurs de poids afin de rendre le problème élémentaire et dépendant uniquement des largeurs des noyaux. Notons que cette méthode ne requiert que deux noyaux pour fonctionner, et ne tire donc pas profit du nombre de noyaux mis à sa disposition. Il a tout de même l'avantage de fonctionner pour tout $N \geq 2$. Si l'on ne fixe pas la valeur du poids comme nous l'avons fait, le paramètre de largeur varie en fonction de celui-ci et de la largeur de bande désirée selon les équations déterminées par L'Espérance (L'Espérance, 2008).

Il y a donc en fait une infinité de paires de paramètres (w, S) qui produisent la bande de largeur B. Ceci dit, nous allons conserver les équations simplifiées qui nous permettent de facilement ajouter une interaction contrôlée entre les signaux.

9.2 Les bandes d'interaction

Le plus simple effet sur une image résultant d'une interaction entre les signaux est l'apparition d'une seconde bande blanche au bas ou à droite de l'image. Celle-ci apparaît lorsque l'entrée du signal au délai nul devient suffisamment grande pour que la fin de ce signal arrive à contribuer au maximum du signal suivant et à activer la sortie. Ceci est vrai pour toute interaction provenant de deux signaux positifs. Pour que le résultat soit une simple bande, il est nécessaire que la contribution du premier signal soit suffisante pour l'activation de la sortie, qu'elle continue à l'être lorsque la première entrée augmente et ce peu importe les changements encourus au second signal. Pour ce faire, le plus simple est encore une fois d'utiliser des noyaux dont la hauteur est

très proche du seuil. Comme la position du premier noyau d'un signal ne varie pas en fonction de l'entrée, l'apparition de la bande ne varie alors pas selon l'entrée du signal décalé puisque la contribution du premier signal doit se faire à un point fixe, soit le sommet du premier noyau du second signal. Il est alors possible de contrôler l'épaisseur de la bande d'interaction à l'aide du paramètre de délai θ_y. Le seuil est franchi si la fin du dernier noyau du premier signal atteint le centre du premier noyau du second signal. Il est aussi franchi si c'est le début qui est atteint par le centre, selon lequel des signaux possède les noyaux les plus larges. Ceci ce transforme en :

$$x(N-1) + 2S_x \geq \theta_y + S_y \quad \text{si } S_x > S_y$$
$$x(N-1) + S_x \geq \theta_y \quad \text{si } S_x < S_y$$

ou autrement

$$\theta_y \leq x(N-1) + 2S_x - min(S_x, S_y)\,. \tag{9.5}$$

Alors, en fonction de la largeur B en pixels de la bande désirée et du nombre de pixels L de l'image :

$$\theta_y \leq \frac{L-B+0{,}25}{L}(N-1) + 2S_x - min(S_x, S_y) \tag{9.6}$$

où nous avons cette fois ajouté un quart de pixel à la bande.

Nous avons ainsi les équations pour une image des bandes de largeurs B_{y1} à gauche, B_{x1} en haut et B_{x2} en bas, et ce, avec deux entrées. Afin de compléter et fermer un rectangle de grosseur variable, nous devons ajouter une quatrième bande à la droite de l'image. Pour ce faire, il est nécessaire d'ajouter un troisième signal, et donc de doubler l'une des entrées. L'une ou l'autre des entrées est adéquate pour la tâche à accomplir, prenons X. L'objectif de ce troisième signal est de permettre au second signal, soit Y, de produire une activation lorsque celui-ci s'étend et parvient à l'atteindre. Les équations qui régissent ses paramètres sont simplifiées en conséquence. D'abord, afin d'éliminer la légère ambiguïté que représente le terme $min(S_{x2}, S_y)$ dans l'équation du délai, nous fixons la largeur des noyaux du nouveau signal à $S_{x2} = 1$. Comme les largeurs du signal Y ne dépassent jamais $(19 + 0{,}5)/20 = 0{,}975$ pour $L = 20$ ou même

$\lim_{L\to\infty}(L-1+0{,}5)/L=1$, le terme $min(S_{x2},S_y)=S_y$. Nous pouvons alors avoir l'équation du délai du nouveau signal :

$$\theta_{x2}\leq\theta_y+\frac{L-B+0{,}25}{L}(N-1)+S_y. \tag{9.7}$$

Ensuite, comme la largeur de la première bande horizontale du haut est déjà régie par le premier signal X, nous ne voulons pas que le nouveau venu arrive à le supplanter. Nous pouvons nous en assurer en ne permettant pas au signal X_2 d'activer la sortie pour toute valeur d'entrée. Supposons possible la plus petite valeur d'entrée $x=0$, les N noyaux du signal sont alors superposés et le maximum résultant est $max=Nw_{x2}$. Comme nous désirons que ce maximum n'active pas la sortie, et ne parvienne donc pas au seuil, nous avons la condition :

$$Nw_{x2}\leq\Gamma$$

ou bien

$$w_{x2}\leq\frac{\Gamma}{N}. \tag{9.8}$$

En réalité, dans nos images les coordonnées des pixels sont toujours supérieures à zéro, ainsi nous pouvons prendre l'égalité et respecter notre condition.

En somme, nous avons les équations pour tous les paramètres d'un quantron XXY qui produit l'image d'un rectangle noir de taille et position déterminées par les bandes blanches qui le construisent. Pour les bandes du haut et du bas respectivement de tailles $B_{x1}\ et\ B_{x2}$ puis les bandes de gauche et de droite de tailles $B_{y1}\ et\ B_{y2}$:

$$\begin{pmatrix} w_{x1}\\ S_{x1}\\ \theta_{x1}\\ w_y\\ S_y\\ \theta_y\\ w_{x2}\\ S_{x2}\\ \theta_{x2}\end{pmatrix}=\begin{pmatrix} \Gamma-d\\ (B_{x1}+0{,}5)/L\\ 0\\ \Gamma-d\\ (B_{y1}+0{,}5)/L\\ (N-1)(L-B_{x2}+0{,}25)/L+2S_{x1}-min(S_{x1},S_y)\\ \Gamma/N\\ 1\\ \theta_y+(N-1)(L-B_{y2}+0{,}25)/L+S_y\end{pmatrix}. \tag{9.9}$$

Voici par exemple la création de la lettre O en spécifiant des bandes de six pixels.

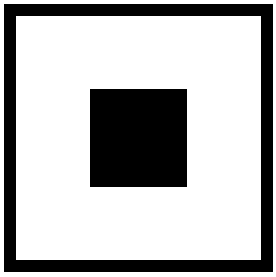

Figure 9.2 : Image de la lettre O par rectangle aux bandes B_{x1}, B_{x2}, B_{y1} et $B_{y2} = 6$

Ses paramètres sont les suivants.

$$\begin{pmatrix} w_{x1} \\ S_{x1} \\ \theta_{x1} \\ w_y \\ S_y \\ \theta_y \\ w_{x2} \\ S_{x2} \\ \theta_{x2} \end{pmatrix} = \begin{pmatrix} 1 - 0.0001 \\ (6 + 0{,}5)/20 \\ 0 \\ 1 - 0.0001 \\ (6 + 0{,}5)/20 \\ (10 - 1)(20 - 6 + 0{,}25)/20 + 2 \times 6 - min(S_{x1}, S_y) \\ 1/10 \\ 1 \\ \theta_y + (10 - 1)(20 - 6 + 0{,}25)/20 + S_y \end{pmatrix} = \begin{pmatrix} 0{,}9999 \\ 0{,}3250 \\ 0 \\ 0{,}9999 \\ 0{,}3250 \\ 6{,}6500 \\ 0{,}1000 \\ 1{,}0000 \\ 13{,}3000 \end{pmatrix} \quad (9.10)$$

Si l'une ou l'autre des bandes B_{x2}et B_{y2} sont nulles, il n'est pas nécessaire d'ajouter la troisième entrée. Il suffit de s'assurer que l'entrée au délai nul soit celle qui requiert une seconde bande. Notons aussi que pour assurer une meilleure réponse, il peut être nécessaire d'interchanger les formules des signaux X et Y selon lequel possède la première bande B_{x1} ou B_{y1} la plus étroite.

9.3 Comparaisons avec le perceptron

Nous avons donc, pour les deux dimensions $X\ et\ Y$ d'une image, un total de neuf paramètres nécessaires à la formation d'un rectangle quelconque et ceci avec un seul quantron. Pour un résultat équivalent, le MLP requiert l'utilisation d'un perceptron pour former chaque côté du rectangle et un perceptron supplémentaire à la couche suivante pour assembler la forme. Les

premiers perceptrons ne requièrent qu'une entrée et donc un poids en plus du paramètre de biais. Le perceptron de sortie a autant d'entrées qu'il y a de bandes. Le MLP équivalent possède ainsi cinq perceptrons et un total de treize paramètres. Supposons qu'une dimension supplémentaire est ajoutée au problème de sorte que c'est maintenant un prisme rectangulaire qui doit être formé. Pour résoudre le problème, le MLQ à quantron unique n'a qu'à ajouter une entrée supplémentaire qui dépend de la nouvelle dimension, portant alors le nombre de paramètres à douze et le nombre de quantrons restant un. Le MLP doit ajouter deux autres perceptrons à la couche d'entrée et deux entrées supplémentaires au perceptron de la couche de sortie, portant le nombre de paramètres à 19 et le nombre de perceptrons à 7. Si le nombre de dimensions N_D du problème continue à augmenter, les nombres de paramètres et de quantrons ou perceptrons peuvent être obtenus par les formules suivantes.

$$\begin{gathered}\text{Nombre de paramètres du MLQ} = 3N_D + 3 \\ \text{Nombre de quantrons} = 1 \\ \text{Nombre de paramètres du MLP} = 6N_D + 1 \\ \text{Nombre de perceptrons} = 2N_D + 1\end{gathered} \tag{9.11}$$

Nous voyons que même pour une seule dimension, le MLQ est avantagé sur le nombre de paramètres et d'éléments nécessaires. Alors que le nombre de dimensions augmente, le nombre de paramètres du MLQ tend à s'approcher de la moitié du même nombre pour le MLP équivalent. De plus, le nombre de quantrons nécessaires reste toujours un alors que le nombre de perceptrons augmente linéairement.

Nous venons de montrer l'avantage du MLQ par rapport au MLP pour la création de prismes rectangulaires à plusieurs dimensions. Supposons maintenant qu'il est nécessaire de former une certaine région, ou une multitude de régions, dans un hyperespace de N_D dimensions à partir de la combinaison d'une quantité k de tels prismes. C'est une relation logique de type « ET » qu'il faut appliquer aux rectangles pour obtenir l'effet désiré. Malheureusement, il faut alors ajouter une couche supplémentaire pour effectuer l'opération, alors qu'une relation « OU » aurait pu être faite à l'intérieur même d'un seul quantron. Chaque prisme doit être produit par un quantron et la combinaison effectuée par un dernier quantron à la sortie. Nous avons alors :

$$\text{Nombre de paramètres du MLQ} = 3kN_D + 6k$$
$$\text{Nombre de quantrons} = k + 1$$
$$\text{Nombre de paramètres du MLP} = 6kN_D + 2k + 1$$
$$\text{Nombre de perceptrons} = 2kN_D + k + 1. \quad (9.12)$$

Bien qu'il garde encore un avantage sur le MLP, nous voyons ici que les trois paramètres nécessaires à chaque entrée du quantron augmentent le nombre de paramètres du réseau inutilement lorsqu'utilisés uniquement pour effectuer des relations logiques « ET ». Il serait envisageable de créer un réseau mixte utilisant la force du quantron pour la formation des premières images et la simplicité du perceptron pour les combiner.

Nous avons, à l'aide de l'expérience acquise sur les signaux du quantron, proposé une méthode simple pour déterminer les paramètres nécessaires à la formation de rectangles quelconques. En agençant une quantité suffisante de ces rectangles, il est possible en théorie de produire n'importe quelle surface discriminante désirée. Il s'agit finalement d'une preuve que le MLQ peut effectivement représenter exactement un ensemble d'entraînement particulier, mais seulement si le nombre de quantrons et de paramètres respecte les équations (9.12). Celles-ci sont plus avantageuses que pour l'équivalent MLP, mais un hybride serait encore mieux afin de tirer avantage des forces de chaque modèle. Dans la prochaine section, nous déterminons les paramètres nécessaires au quantron pour effectuer les différentes opérations logiques requises pour les images de lettres produites.

10 OPÉRATIONS LOGIQUES

Lorsqu'une lettre est formée à l'aide de plusieurs images produites par différents quantrons, il est nécessaire de les combiner. Les sorties étant binaires, ces combinaisons sont en fait des opérations logiques booléennes. Une ou plusieurs couches de quantrons supplémentaires peuvent s'occuper de cette tâche, c'est leurs paramètres que nous déterminons ici.

10.1 Inspiration du perceptron

Les relations logiques entre les images nécessaires pour notre application sont relativement simples. Nous avons déjà montré comment la relation « OU » peut être incorporée directement à l'intérieur du quantron qui produit les deux images à combiner. Il demeure requis d'effectuer l'inversion binaire et la relation « ET » pour certaines lettres telles le J et le A. Ces opérations s'effectuent facilement à l'aide de simples perceptrons et c'est à partir de ces derniers que nous nous inspirons pour déterminer les paramètres des quantrons qui font le travail.

Il est en fait possible, en fixant certains paramètres, de transformer un quantron en perceptron lorsqu'il s'agit d'entrées binaires. En fixant les valeurs des largeurs $S = 1$, pour une entrée active « =1 », chaque noyau débute au sommet du noyau précédent. L'on se retrouve alors avec un signal de la forme d'un plateau de hauteur w. Une entrée non active « =0 » est interprétée comme une absence de signaux. Si l'on fixe ensuite les délais à zéro pour tous les signaux, la hauteur du signal total est simplement la somme des w pour lesquels les entrées sont actives, ou le produit scalaire entre le vecteur de poids $\overrightarrow{W}$ et le vecteur d'entrées $\vec{X}$. Avec un seuil fixé à $\Gamma = 1$, on se retrouve au bout du compte avec l'équivalent d'un perceptron à fonction de transfert échelon $H(x - 1)$, ou bien $H(x)$ avec biais de -1.

10.2 Paramètres

Une fois l'outil trouvé, il suffit de déterminer les poids qui permettent d'obtenir les opérations logiques nécessaires aux lettres produites.

- Pour la relation $A \cap B$ requise par les lettres T, E, N et M il suffit d'utiliser $\overrightarrow{W} = (0{,}75,\ 0{,}75)$.
- Pour la relation $\bar{A}$ requise par la lettre J et I il suffit d'utiliser $\overrightarrow{W} = (-1,\ 1{,}5)$ en fixant la valeur de la deuxième entrée à 1.
- Pour la relation $\bar{A} \cup B$ requise par les lettres H et U, il est possible d'utiliser $\overrightarrow{W} = (-1,\ 1{,}5,\ 1{,}5)$ en fixant la troisième entrée à 1 et le deuxième délai à 10. Ceci tire avantage de l'opération « OU » interne du quantron.
- Pour la relation $\bar{A} \cap B \cap C$ requise par la lettre A, il suffit d'utiliser $\overrightarrow{W} = (-1,\ 0{,}75,\ 0{,}75$.

Ces quatre relations suffisent pour les images de lettres que nous avons produites. Il serait possible de continuer à déterminer les paramètres de toute autre relation logique nécessaire.

Remarquons que grâce à l'opération « OU » interne du quantron, n'importe quelle opération logique peut être effectuée par un seul quantron tant que suffisamment d'entrées lui sont fournies. Il suffit alors de décomposer l'opération en unions d'intersections puis de traiter chaque groupe d'intersections avec des signaux séparés par des délais suffisamment grands. Nous avons déjà remarqué l'inconvénient des trois paramètres nécessaires pour chaque entrée du quantron lorsque l'on désire effectuer des opérations logiques simples, telles une série de « ET » dans la section précédente. Nous voyons par contre que pour des opérations logiques plus complexes, le quantron tire alors profit de son fonctionnement non linéaire et peut arriver à simplifier significativement un réseau.

11 DIAGONALES ET COURBES

Un des principaux avantages qu'a le quantron est sa capacité à produire des régions non linéaires, ce qu'un seul perceptron n'est pas capable de faire. Cette propriété est utilisée dans la production de toutes les images de lettres proposées. Nous avons d'ailleurs automatiquement rejeté les images aléatoires produites qui ne comportaient qu'une ligne droite. Par contre, parmi les lettres produites, seule la lettre J inclut un semblant de courbe. Or, la capacité à produire des courbes serait un avantage significatif par rapport à un perceptron qui doit pour les approximer utiliser la combinaison d'un certain nombre de lignes droites. Notons aussi la relative rareté des diagonales dans les lettres produites. Seules les lettre A et N les utilisent et ce sont toutes des diagonales orientées de la même façon. Ces observations mènent à la présente section qui tente d'expliquer le manque de courbes utiles et de variété de diagonales dans les images produites.

11.1 Diagonales et courbes négatives

Nous venons d'observer que seulement deux lettres utilisent des diagonales, et que celles-ci sont orientées de la même façon. Nous désignons cette orientation « positive » puisqu'elle s'apparente à une droite à pente positive lorsque les valeurs X et Y augmentent. Par opposition, nous désignons « négatives » les diagonales d'orientation contraire. Notons immédiatement que l'orientation de la diagonale est invariante à l'inversion des axes X et Y de l'image. Or, le manque de diagonales négatives observé dans l'ensemble des lettres produites est aussi observé dans le plus grand ensemble des images aléatoires. Alors que celui-ci contient une grande variété de diagonales positives de différentes grandeurs et orientations, nous trouvons très peu de diagonales négatives, qui sont courtes et tendent à être courbes.

Pour tenter d'expliquer le manque de ce type de diagonales, nous montrons ici qu'elles sont impossibles à obtenir pour au moins 50% des configurations de quantrons concevables. Soient les deux diagonales négatives suivantes.

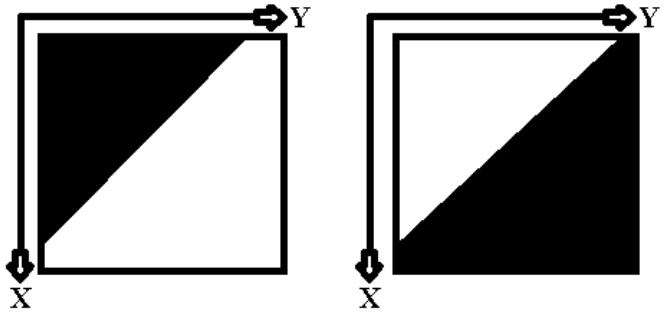

Figure 11.1 : Images de diagonales négatives

Ces images ne représentent pas nécessairement l'entièreté de la région $0 < X \leq 1, 0 < Y \leq 1$ normalement utilisée et peuvent n'être qu'un petit élément de celles-ci. La convention des axes est toutefois respectée, soit X augmente du haut vers le bas et Y de la gauche vers la droite. Bien que les deux diagonales aient la même pente, si l'on obtient des paramètres qui les produisent, il suffit d'utiliser la propriété du deuxième lemme sur les paramètres pour étirer un des axes et modifier la pente par le fait même. Ainsi, prouver sous certaines conditions qu'il est impossible d'obtenir ces diagonales implique que, sous les mêmes conditions, aucune image produite par le quantron ne possède de partie à diagonale négative. Les courbes correspondantes sont alors aussi impossibles, car la diagonale peut être considérée comme une partie infiniment petite de ces courbes.

Nous distinguons quatre cas particuliers. En premier lieu, les deux images ci-dessus doivent être considérées pour tenir compte des passages d'actif à inactif et vice versa. Ensuite, il y a les quatre combinaisons de signaux X et Y positifs et négatifs. De ces quatre derniers, nous pouvons déjà éliminer le cas (X^-, Y^-), soit les deux signaux négatifs, puisqu'ils ne produisent que des images noires. Considérant la propriété d'invariance du type de diagonale par rapport à l'inversion des axes, ce qui est vrai pour le cas (X^+, Y^-) l'est aussi pour le cas (X^-, Y^+). Nous nous retrouvons avec seulement quatre cas, dont deux d'entre eux peuvent être démontrés comme impossibles et un cas rare et non pratique.

11.1.1 Cas #1 : (X^+Y^+) avec passage d'inactif à actif

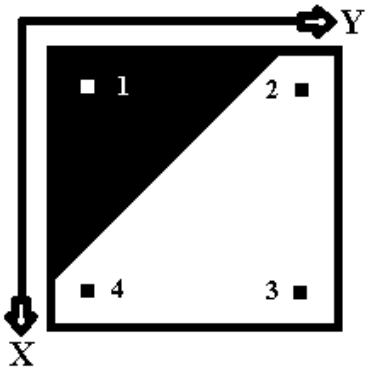

Figure 11.2 : Positions de points de référence sur une diagonale négative pour le cas #1

Nous avons indiqué sur l'image l'emplacement de quatre points distincts. Les coordonnées exactes de ces points ne sont pas importantes, c'est plutôt leurs positions relatives les unes aux autres qui jouent un rôle dans la démonstration. Pour passer du point 1 au point 2, seule la valeur de l'entrée Y augmente. Pour passer de 2 à 3, seul X augmente. Pour 3 à 4, Y diminue puis pour 4 à 1, c'est X qui diminue. Notons que pour un signal donné, lorsque la valeur de son entrée augmente, ce signal tend à s'étendre et à diminuer en amplitude. Au contraire, lorsque son entrée diminue, il tend à se raccourcir et à augmenter en amplitude.

Nous pouvons ainsi qualifier les signaux en fonction des différents points.

1. X court et haut, Y court et haut.
2. X court et haut, Y long et bas.
3. X long et bas, Y long et bas.
4. X long et bas, Y court et haut.

Prenons comme premier cas possible avec cette image des signaux (X^+, Y^+) tous deux positifs. Pour obtenir cette image il est nécessaire que ces deux conditions soient respectées.

1. Il y a une transition d'inactif à actif en passant du point #1 au point #2.
2. Il y a une transition d'inactif à actif en passant du point #1 au point #4.

En passant du point #1 au point #2, le signal X reste inchangé. Le signal Y lui s'allonge et perd en amplitude. Pour que ceci mène à une transition d'inactif à actif, cela implique que le signal Y en s'allongeant parvient à rejoindre d'avantage le signal Y et à s'y additionner pour franchir le seuil et donc qu'au point #1, le sommet du signal Y se trouve avant celui du signal X.

En passant du point #1 au point #4, les signaux sont interchangés et c'est alors le sommet du signal X qui doit débuter avant celui du signal Y au point #1. Comme les deux cas ne peuvent être vrais en même temps, les conditions 1 et 2 sont mutuellement exclusives et par conséquent l'image ne peut être produite.

11.1.2 Cas #2 : (X^+, Y^-) avec passage d'inactif à actif

Prenons maintenant le cas des signaux (X^+, Y^-). Pour obtenir l'image légèrement modifiée :

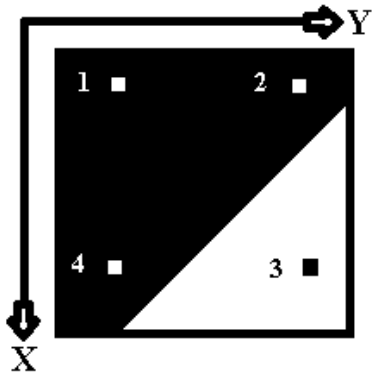

Figure 11.3 : Positions des points de référence sur une diagonale négative pour le cas #2

il est nécessaire que ces quatre conditions soient respectées.

1. Le signal X^+ parvient à atteindre le seuil par lui-même, et ce, même s'il est bas puisque le point # 3 est actif.
2. Au point #2, un signal Y^- long et bas parvient à atténuer un signal X^+ court et haut.
3. Au point #3, un signal Y^- long et bas ne parvient pas à atténuer un signal X^+ long et bas.
4. Au point #4, un signal Y^- court et haut parvient à atténuer un signal X^+ long et bas.

Selon la condition 2, l'amplitude basse du signal Y^- suffit à atténuer l'amplitude haute du signal X^+. Selon la condition 3, cette même amplitude ne parvient pas à atténuer l'amplitude basse du

signal X^+. Ceci est possible seulement si, au point 3, le sommet du signal Y^- précède celui du signal X^+ et qu'en s'allongeant, ce dernier sort de l'influence du premier.

Or, selon la condition 4, le signal Y^- court atténue le signal X^+ long, qui bien que bas, atteint le seuil à lui seul selon la condition 1. Selon la condition 3, le même signal X^+ long n'est pas atténué par un signal Y^- cette fois long. Ceci est possible seulement si, au point 3, le sommet du signal Y^- se trouve après celui du signal X^+ et qu'en s'allongeant, se retire et cesse de l'atténuer suffisamment. Encore une fois, les signaux ne peuvent à la fois se trouver avant et après l'un l'autre au même point et par conséquent l'image ne peut être produite.

Ainsi, il n'est pas possible de produire une diagonale à pente négative avec un passage d'inactif à actif lorsque les entrées augmentent. Voyons maintenant pour le cas d'un passage d'actif à inactif.

11.1.3 Cas #3 : (X^+, Y^-) avec passage d'actif à inactif

Soit l'image inverse du cas #1 avec les quatre points.

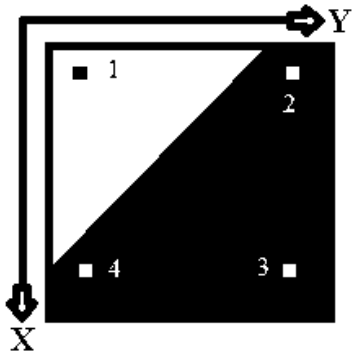

Figure 11.4 : Positions de points de référence sur une diagonale négative inversée

Prenons d'abord le cas des signaux (X^+, Y^-). Pour obtenir l'image il est nécessaire que ces trois conditions soient respectées.

1. Par le point #1, le signal Y^- n'empêche pas le signal X^+ de produire l'activation pour une partie de l'image.
2. Au point #2, un signal Y^- long et bas parvient à atténuer un signal X^+ court et haut.
3. Au point #4, un signal Y^- court et haut parvient à atténuer un signal X^+ long et bas.

Ceci est en fait possible. Si seulement le signal X^+ haut est apte à atteindre le seuil, les points 3 et 4 seront bel et bien inactifs. Ensuite, avec un signal Y^- qui précède X^+, il peut alors l'atteindre en s'allongeant lorsque l'entrée Y augmente. Ceci a par contre tendance à ne produire que de très courtes parties diagonales. Le signal X^+ qui n'arrive plus à activer la sortie à lui seul crée une délimitation horizontale et l'atténuation du signal Y^- tend rapidement être trop forte. Un exemple type de ces courtes diagonales est celui obtenu dans la lettre J. Nous inversons ici les axes pour respecter l'ordre des signaux précédents.

Figure 11.5 : Courte diagonale négative présente dans la lettre *J*

Donc bien que possible, ce cas est rare et peu pratique pour la formation de diagonales plus longues.

11.1.4 Cas #4 : (X^+, Y^+) avec passage d'actif à inactif

Le cas d'une transition d'actif à inactif avec des signaux positifs (X^+, Y^+) s'avère être la seule alternative abordable. Deux signaux positifs qui arrivent à se superposer pour atteindre le seuil peuvent tous les deux faire diminuer le maximum de l'interaction en s'allongeant et en diminuant leur propre amplitude. Ceci crée par contre des formes courbes plutôt que droites, car les deux signaux contribuent à l'atténuation du maximum lorsqu'ils diminuent. Voici tois exemples de ces courbes obtenues dans les images aléatoires produites.

Figure 11.6 : Exemples de courbes négatives du deuxième cadran

En somme, nous avons prouvé qu'il n'est pas possible de produire des diagonales ou courbes négatives avec un passage d'inactif à actif. Dans le cas d'un passage d'actif à inactif, seules des courbes très localisées peuvent être produites pour le quatrième cadran du cercle. Il demeure cependant possible de produire des courbes du deuxième cadran de différentes grandeurs.

Ces analyses sont faites en ne considérant que des entrées simples et un seul quantron. Comme nous l'avons montré plus tôt, un seul quantron à la couche suivante d'un MLP suffit à inverser les pixels de l'image et donc d'obtenir le passage d'inactif à actif si désirer. Aussi, il est possible que les limites puissent être contournées à l'aide de l'ajout d'entrées multiples. Nous observons d'ailleurs dans les images aléatoires produites quelques rares exemples qui tendent à nous le suggérer.

Figure 11.7 : Diagonales négatives possibles avec des entrées multiples

11.2 Diagonales et courbes positives

Pour les diagonales positives, un changement des axes produit une inversion du passage d'actif à inactif vers un passage d'inactif à actif et nous arrivons à les produire avec différents angles et longueurs, dont voici quelques exemples.

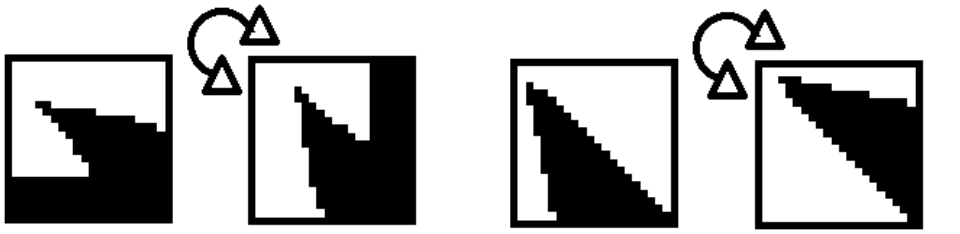

Figure 11.8 : Différentes images de diagonales positives

Nous en avons d'ailleurs fait usage dans les lettres A et N.

Pour les courbes, l'inversion des axes produit un changement d'orientation de la courbe. Nous arrivons à trouver quelques éléments courbes dans les images aléatoires produites.

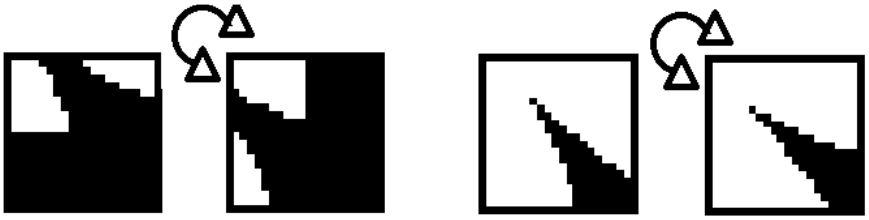

Figure 11.9 : Différentes images de courbes positives du premier et troisième cadran

Notons la similitude avec les images de diagonales présentées juste avant, ce sont en fait de légères modifications des paramètres qui font passer des unes aux autres. Bien que possibles, les courbes obtenues ne sont disponibles qu'en sections et accompagnées d'autres éléments indésirables. Pour les utiliser à la fabrication de lettres, il est nécessaire de ne sélectionner que certaines parties spécifiques des images et d'en assembler une grande quantité. Le but premier du travail n'étant pas de produire des lettres parfaites à tout prix, mais d'étudier le quantron, nous avons laissé de côté cette possibilité. Nous avons plutôt utilisé les équations de la section 9 pour fabriquer les lettres manquantes.

Chapitre 3 : RÉSULTATS

12 LES 26 LETTRES DE L'ALPHABET

Tout au long du travail d'exploration, nous avons utilisé comme fil conducteur la production d'images de caractères de l'alphabet. Ceci nous a permis d'observer les comportements du modèle du quantron dans le cadre particulier de la création de surfaces discriminantes nécessaires à la résolution de problèmes de classification. Aux quelques lettres obtenues au départ se sont graduellement ajoutées de nouvelles alors qu'étaient découvertes les propriétés du quantron. Nous nous retrouvons avec un ensemble de 14 lettres aux formes variées dont nous connaissons les architectures et les paramètres des MLQs qui les produisent. Celles-ci ont été raffinées afin d'en améliorer l'aspect et de nous familiariser avec les signaux du quantron.

Il manque près de la moitié des caractères de l'alphabet à notre banque. Ceux-ci s'avèrent être difficiles à produire par l'assemblage d'une petite quantité d'images compte tenu du manque de courbes et diagonales exposé à la section précédente. Il serait en effet nécessaire de découper en sections et de manipuler une plus grande quantité d'images pour les obtenir de cette façon ce qui serait très long et sans grand intérêt. Nous avons toutefois démontré à la section 9 que la production de n'importe quelle surface est possible, et ce, avec la précision désirée tant que suffisamment de quantrons sont utilisés. Nous y fournissons également les équations qui permettent de calculer les paramètres nécessaires. À titre d'exemple, nous montrons ici la production de la lettre D par assemblage de rectangles.

Un premier niveau de détails de la lettre D peut être formé des quatre rectangles suivants.

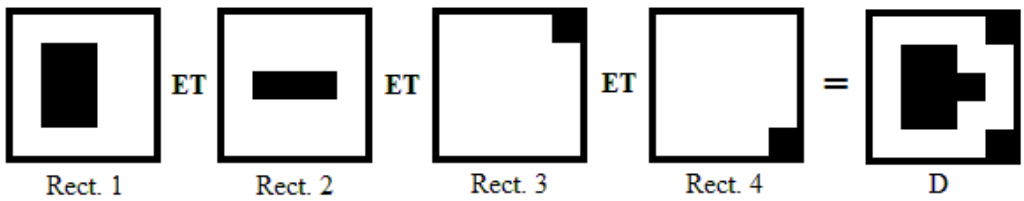

Figure 12.1 : Formation de la lettre D par assemblage de rectangles

Les dimensions des bandes pour chacun des 4 rectangles sont :

- Rect.1: $B_{x1} = 4, B_{x2} = 4, B_{y1} = 4, B_{y2} = 8.$

- Rect.2: $B_{x1} = 8, B_{x2} = 8, B_{y1} = 4, B_{y2} = 4.$

- Rect.3: $B_{x1} = 0, B_{x2} = 16, B_{y1} = 16, B_{y2} = 0.$

- Rect.4: $B_{x1} = 16, B_{x2} = 0, B_{y1} = 16, B_{y2} = 0.$

Avec ces dimensions et les équations déterminées à la section 9, nous pouvons calculer les paramètres des quatre quantrons de la couche d'entrée qui permettent de former cette lettre.

$$\begin{pmatrix} w_{x1,1} \\ S_{x1,1} \\ \theta_{x1,1} \\ w_{y1} \\ S_{y1} \\ \theta_{y1} \\ w_{x1,2} \\ S_{x1,2} \\ \theta_{x1,2} \end{pmatrix} = \begin{pmatrix} 0{,}9999 \\ 0{,}2250 \\ 0 \\ 0{,}9999 \\ 0{,}2250 \\ 7{,}5375 \\ 0{,}1000 \\ 1 \\ 13{,}2750 \end{pmatrix}, \begin{pmatrix} w_{y2,1} \\ S_{y2,1} \\ \theta_{y2,1} \\ w_{x2} \\ S_{x2} \\ \theta_{x2} \\ w_{y2,2} \\ S_{y2,2} \\ \theta_{y2,2} \end{pmatrix} = \begin{pmatrix} 0{,}9999 \\ 0{,}2250 \\ 0 \\ 0{,}9999 \\ 0{,}4250 \\ 7{,}5375 \\ 0{,}1000 \\ 1 \\ 13{,}4750 \end{pmatrix},$$

$$\begin{pmatrix} w_{x3} \\ S_{x3} \\ \theta_{x3} \\ w_{y3} \\ S_{y3} \\ \theta_{y3} \end{pmatrix} = \begin{pmatrix} 0{,}9999 \\ 0{,}0250 \\ 0 \\ 0{,}9999 \\ 0{,}8250 \\ 1{,}9375 \end{pmatrix}, \begin{pmatrix} w_{x4} \\ S_{x4} \\ \theta_{x4} \\ w_{y4} \\ S_{y4} \\ \theta_{y4} \end{pmatrix} = \begin{pmatrix} 0{,}9999 \\ 0{,}8250 \\ 0 \\ 0{,}9999 \\ 0{,}8250 \\ 9{,}9375 \end{pmatrix} \qquad (12.1)$$

Les paramètres du quantron de sortie qui effectue la relation OU entre les quantrons sont fixés selon la logique exposée à la section 10.

$$\begin{pmatrix} w_{s1} \\ S_{s1} \\ \theta_{s1} \\ w_{s2} \\ S_{s2} \\ \theta_{s2} \\ w_{s3} \\ S_{s3} \\ \theta_{s3} \\ w_{s4} \\ S_{s4} \\ \theta_{s4} \end{pmatrix} = \begin{pmatrix} 0{,}2600 \\ 1 \\ 0 \\ 0{,}2600 \\ 1 \\ 0 \\ 0{,}2600 \\ 1 \\ 0 \\ 0{,}2600 \\ 1 \\ 0 \end{pmatrix} \tag{12.2}$$

Avec les paramètres déterminés, l'architecture du MLQ qui produit l'image de la lettre *D* est donnée à la figure suivante.

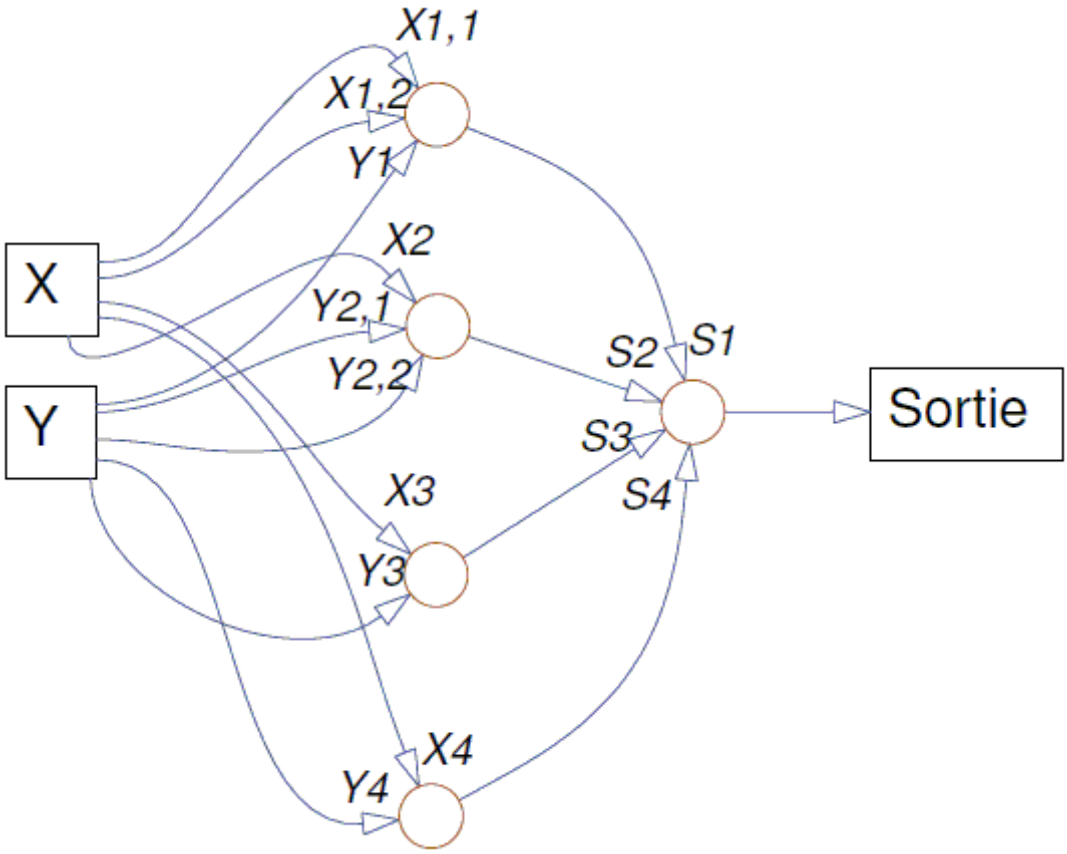

Figure 12.2 : Architecture du MLQ produisant l'image de la lettre *D*

Il est possible ensuite de raffiner la lettre en ajoutant des rectangles supplémentaires. Par exemple, les images suivantes sont obtenues en greffant respectivement 3 et 10 quantrons au réseau.

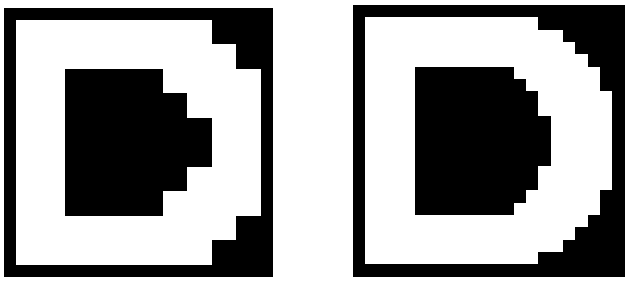

Figure 12.3 : Deux niveaux supérieurs de raffinement de la lettre D

La même procédure peut être faite pour les 11 autres caractères manquants ou pour toute autre forme désirée. Notons que nous avons aussi diminué le nombre de quantrons utilisé pour les lettres O et U à l'aide de l'assemblage de rectangles. Ainsi, nous sommes en mesure de compléter la banque de caractères que voici.

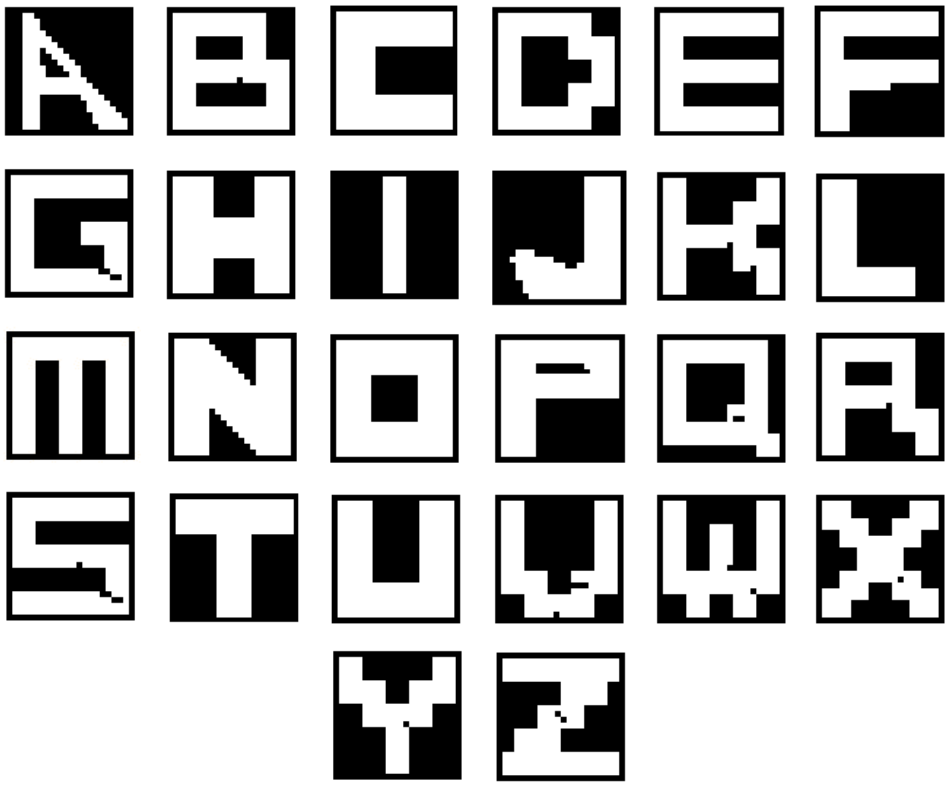

Figure 12.4 : Images des 26 lettres de l'alphabet

Pour la construction par rectangles, nous avons tenté d'assembler les lettres avec le minimum de quantrons permettant d'obtenir une forme reconnaissable. Tout comme nous l'avons fait pour la lettre D, il est possible de raffiner les résultats à l'aide de quantrons supplémentaires.

On peut observer que quelques pixels isolés sont présents sur les images produites par construction avec les rectangles. Ceci est causé par les équations qui déterminent les paramètres de largeurs de signaux. Pour des bandes B_{x1} et B_{y1} petites, voir nulles, les noyaux sont proportionnellement petits. Ceux-ci requièrent alors des précisions d'approximation plus élevées. Il est aussi possible, quand les deux bandes sont petites, que certains pixels pour lesquels les valeurs d'entrées doivent produire une activation par interaction restent noirs parce que les signaux s'intercalent sans s'additionner. Pour régler ce problème, il est envisageable d'ajouter une entrée supplémentaire ou de modifier les équations pour augmenter les largeurs de noyaux et adapter les hauteurs en conséquence.

13 DISCUSSION ET CONCLUSION

Les images précédentes sont prometteuses, nous avons bel et bien réussi à produire les caractères de l'alphabet uniquement à l'aide de MLQs. Environ la moitié des images ont été obtenues par combinaisons délibérées des résultats obtenus aléatoirement. L'aspect de celles-ci a été amélioré en manipulant les paramètres selon l'étude de leurs effets sur les signaux internes des quantrons. Cette étape importante nous a aidés à nous familiariser avec le fonctionnement du quantron et à en déduire différentes propriétés. Nous en sommes ultimement arrivés à des règles qui nous ont permis de contourner certaines lacunes dans les formes produites aléatoirement et d'obtenir par construction les lettres manquantes. En somme, l'exploration du modèle du quantron dans le cadre de la production d'images de caractères de l'alphabet a fait ressortir plusieurs caractéristiques inédites, avantages et désavantages, que des travaux futurs pourront exploiter et approfondir.

13.1 Avantages du modèle

Le premier avantage découvert est la possibilité pour le quantron de réutiliser plusieurs fois la même valeur d'entrée et d'en obtenir des résultats sans équivalence à une entrée unique. Ceci est possible grâce à sa fonction d'activation complexe et dépendante pour chaque entrée de trois paramètres régissant différents aspects des signaux. Particulièrement, les paramètres de délais θ permettent, si désiré, de totalement séparer différents groupes de signaux et ainsi d'obtenir facilement une relation logique de type OU à l'intérieur même d'un seul quantron.

À la section 10, nous avons montré comment ceci permet de produire toute relation logique désirée entre des entrées binaires. Un seul quantron à la couche de sortie du MLQ permet la combinaison des images produites par ceux de la couche d'entrée. À la section 9, nous avons utilisé notre expérience acquise lors de l'étude des signaux pour proposer des équations permettant de calculer les paramètres qui créent des formes rectangulaires de base. En combinant celles-ci à l'aide du quantron de sortie, il est possible d'obtenir n'importe quelle image désirée, ce que nous avons fait pour les 12 lettres qui nous manquaient. Les mêmes équations pouvant être

étendues à plus de deux dimensions, le quantron s'avère un candidat potentiellement intéressant pour des problèmes de reconnaissance de formes à plusieurs variables.

Moins simple, mais tout de même envisageable, est la relation ET interne au quantron. Une telle opération pourrait être faite entre deux groupes de signaux de la façon suivante : prenant deux groupes qui arrivent normalement à atteindre le seuil pour différentes valeurs d'entrée, diminuer de moitié les paramètres de poids w de toutes les entrées, ensuite, décaler les groupes de sorte que les maximums de leurs signaux soient alignés aux valeurs qui activaient précédemment la sortie. L'alignement pour toutes ces valeurs simultanément peut être impossible sur une seule superposition des signaux, mais grâce au décalage du délai θ, les groupes peuvent être répétés indépendamment autant de fois nécessaires pour obtenir au bout du compte la relation ET interne désirée. Nous ne démontrons pas dans ce travail cette proposition, car la manipulation des signaux est très spécifique, mais nous croyons tout de même en sa faisabilité compte tenu de nos observations lors de leur étude. Avec les deux relations logiques ET et OU possibles à l'interne, ceci signifierait qu'un seul quantron pourrait en théorie produire n'importe quelle surface discriminante tant que suffisamment d'entrées lui sont fournies. Une telle propriété donnerait une grande puissance au quantron, mais c'est au coût d'un certain éloignement de l'origine biologique du modèle et de l'esprit connexionniste des réseaux de neurones.

L'utilisation multiple de la même valeur d'entrée permet aussi une certaine équivalence entre les formes des noyaux utilisés. S'il est possible de construire n'importe quelle forme à l'aide de la superposition d'un nombre suffisant par exemple de triangles, alors allouer ce même nombre d'entrées avec les paramètres appropriés et des noyaux triangulaires serait l'équivalent d'utiliser cette forme comme noyau des signaux du quantron. Ceci nous permet en théorie d'utiliser n'importe quelle forme désirée comme noyau dans nos réseaux.

13.2 Désavantages du modèle

Nous avons montré à la section 11 que les frontières diagonales et les frontières courbes sont plutôt rares, voir impossibles dans certains cas, lorsque chaque valeur d'entrée n'est utilisée qu'une seule fois. Ceci est une déception, car l'on s'attendait à ce que ce soit un grand avantage

du quantron par rapport au perceptron. Il a tout de même été possible de contourner ce problème grâce aux diverses nouvelles propriétés découvertes. Comme nous venons d'exposer, l'utilisation multiple des valeurs d'entrées promet de rendre possible ces formes à la précision voulue tant qu'une quantité suffisante de celles-ci sont utilisées. La méthode d'exploration par production aléatoire d'images ne s'est par contre pas avérée efficace pour plus de deux entrées.

Le nombre élevé de paramètres par entrée permet la production d'une grande variété de signaux et par conséquent de fonctions d'activation, mais ce même nombre a rendu difficile l'exploration de leurs plages respectives. Nous avons proposé certains lemmes qui ont permis de construire des règles réduisant ces plages. Celles-ci ont été efficaces pour les images à deux entrées, mais moins pour l'ajout d'entrées supplémentaires, chacune de celles-ci rendant plus difficile de prévoir les interactions possibles entre les signaux.

Sans doute l'ennui le plus sérieux est le temps que requiert le modèle pour obtenir une réponse. Contrairement au perceptron qui ne requiert qu'une multiplication scalaire entre deux vecteurs, une série d'additions et le calcul d'une fonction d'activation simple, le quantron tel que nous l'avons implémenté requiert plusieurs milliers d'opérations. Ceci rend critiques les limites de précision autant sur l'approximation discrète des signaux que sur le nombre de pixels. Par exemple, le temps requis pour le calcul des images de 400 pixels pour un seul quantron de deux entrées à une précision de 1000 points est d'environ 1 seconde. Bien qu'abordable dans le cadre de ce travail, une application future utilisant un algorithme d'apprentissage itératif, telle la rétropropagation de l'erreur, dans un MLQ de forte taille verrait sont temps de calcul comme une limite handicapante.

13.3 Travaux futurs

Bien des avenues s'offrent à ceux qui choisiraient d'entreprendre des recherches supplémentaires sur le modèle du quantron. L'un des objectifs serait d'améliorer la rapidité de la production des sorties. Ceci est primordial avant de penser pouvoir obtenir une application réelle de reconnaissance de formes avec un MLQ de grande taille. Bien sûr, il est toujours possible d'utiliser des ressources informatiques plus puissantes, mais ceci n'est pas une réelle solution. Il

serait plus efficace d'améliorer la procédure de calcul elle-même. Des travaux antérieurs ont déjà proposé des approximations du modèle dans le but de rendre un apprentissage par rétropropagation de l'erreur possible (J.-F. Connolly, 2005; J. F. Connolly & Labib, 2009; De Montigny, 2007; Lastere, 2005; Pepga Bissou, 2007). Il est envisageable de comparer les résultats qu'offrent ces approximations aux résultats et propriétés du quantron exposés dans le présent travail. Dans le cas où ceux-ci sont suffisamment les mêmes, l'approximation pourrait représenter une forte amélioration de la rapidité de calcul. Ultimement, l'idéal serait d'arriver à concevoir une implémentation matérielle du modèle. La lenteur du calcul de sortie serait alors compensée par le traitement massivement parallèle des quantrons au travers du MLQ tout comme le fait le cerveau. Le nombre d'éléments et de connexions possibles ne serait plus limité par le calcul essentiellement sériel qu'effectuent les processeurs des ordinateurs et de très gros réseaux pourraient être construits.

Nous n'avons, dans le cadre de ce travail, utilisé que la sortie binaire du quantron. Le modèle du quantron propose également une sortie analogue. Tout un travail d'exploration reste donc à faire pour découvrir les nouvelles caractéristiques qui émergent de l'utilisation de cette sortie. Les couches cachées d'un MLQ recevant ces sorties en entrées ne s'occuperaient plus seulement d'effectuer des opérations logiques et un plus grand nombre de couches pourrait ainsi être utilisé sans redondances.

Le plus important et sans doute le plus difficile sera d'établir des règles d'apprentissages adaptées au quantron. La rétropropagation de l'erreur a fait ses preuves pour les MLPs et certaines adaptations ont été proposées pour les MLQs mais cette forme d'apprentissage n'est pas biologiquement crédible et difficilement envisageable dans une implémentation matérielle. Nous imaginons plutôt des règles inspirées de la biologie telles que proposées par Donald Hebb (Hebb, 1949). Il s'agirait alors de renforcer ou affaiblir les liens selon les achalandages, de créer aléatoirement de nouveaux liens et d'utiliser des boucles de rétroactions dans les réseaux.

13.4 Conclusion

Ce travail apporte une meilleure appréciation du potentiel du quantron par la compréhension de ses propriétés. Nous avons proposé des règles sur les paramètres qui permettent d'éviter des quantrons qui s'activent, ou non, pour toutes entrées. Ces règles peuvent être prises en compte lors de l'établissement d'algorithmes d'apprentissage afin d'optimiser les changements de paramètres. Des équations ont été développées afin d'assurer la validité des résultats en fonction de la précision de l'approximation utilisée. Celles-ci peuvent établir avant même la production d'images le nombre de points nécessaires de sorte à ne pas ralentir les calculs inutilement. D'autres règles et équations proposées permettent l'établissement des architectures de MLQs et le calcul de leurs paramètres qui arrivent à former n'importe quelle surface discriminante. Ces équations tirent particulièrement profit de l'avantage majeur que possède le quantron par rapport à l'utilisation multiple des valeurs d'entrées. Cet avantage s'avère être la source de fortes propriétés du modèle et permet une réduction significative de la taille des réseaux. En somme, par une exploration originale du modèle dans le cadre de la formation d'images de caractères alphabétiques, nous croyons avoir confirmé son potentiel et validé l'entreprise de recherches supplémentaires à son sujet.

BIBLIOGRAPHIE

Connolly, J.-F. (2005). *Approche multiechelle pour l'approximation de la fonction discriminante d'un reseau de quantrons* (M.Sc.A., École Polytechnique de Montréal, Montréal). (UMI No. MR16769)

Connolly, J. F., & Labib, R. (2009). A Multiscale Scheme for Approximating the Quantron's Discriminating Function. *Neural Networks, IEEE Transactions on, 20*(8), 1254-1266.

De Montigny, S. (2007). *Étude sur l'apprentissage d'une approximation du quantron* (M.Sc.A, École Polytechnique de Montréal, Montréal (Canada)).

Hebb, D. O. (1949). *The organization of Behavior: A Neuropsychological Theory*. New York: Wiley.

Hinton, G. E. (2006). *To Recognize Shapes, First Learn to Generate Images*. Consulté le 2011-06-15, tiré de http://www.cs.toronto.edu/~hinton/absps/montrealTR.pdf.

Kurzweil, R. (2005). The Singularity Is Near. In (pp. 181). New York, New York: Penguin Books.

L'Espérance, P.-Y. (2008). *Étude de la stabilité de la transmission d'informations dans le cerveau humain*. Projet de fin d'étude. École Polytechnique de Montréal. Montréal, Canada.

L'Espérance, P.-Y. (2010). *Modélisation de la transmission synaptique d'un neurone biologique à l'aide de processus stochastiques* (M.Sc.A, Ecole Polytechnique, Montreal (Canada), Montréal).

Labib, R. (1999). New single neuron structure for solving nonlinear problems. *Proceedings of International Conference on Neural Networks* (Vol. 1, pp. 617-620).

Lastere, R. (2005). *Conception de l'algorithme d'apprentissage supervise d'un reseau multicouche de quantrons* (M.Sc.A., École Polytechnique de Montréal, Canada). (UMI No. MR01350)

Maass, W., & Schmitt, M. (1999). On the complexity of learning for spiking neurons with temporal coding. *Information and Computation, 153*, 26-46.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics, 5*, 115-133.

Megiddo, N. (1984). Linear programming in linear time when the dimension is fixed. *Journal of the ACM, 31*(1), 114-127.

Minsky, M. L., & Papert, S. (1969). *Perceptrons; an introduction to computational geometry*. Cambridge, Mass.: MIT Press.

Pepga Bissou, J. (2007). Conception d'un algorithme d'apprentissage pour un reseau de quantrons (Vol. Ph.D.). Montréal: Ecole Polytechnique de Montréal.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review, 56*, 386-408.

Rumelhart, J. L., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagation of errors. *Nature, 323*, 533-536.

Shepherd, G. M., & Koch, C. (1990). Introduction do synaptic circuits. In G. M. Shepherd, (éd.), *The synaptic Organization of the Brain* (pp. 3-31): Oxford University Press.

Turing, A. M. (1936). On computable numbers, with an application do the entscheidungsproblem. In S. Hawking, (éd.), *God created the integers* (pp. 1293-1326). Philadelphia, Pennsylvania: Running Press.

Widrow, B., & Hoff, M. E. (1960). Adaptive switching circuits. *IRE WESCON Convention Record*, 96-104.

ANNEXE 1 – Paramètres

Se trouvent ici les paramètres des MLQs qui produisent les images de certaines figures présentes dans le document. La numérotation des paramètres est telle que présentée dans la liste des sigles et abréviations. Conformément à la section 10, les paramètres de largeurs de noyaux S_S et de délais θ_S des quantrons de sortie sont respectivement 1 et 0 excepté lorsque spécifiés autrement.

Tableau A-1 : Paramètres des MLQs produisant les images de la figure 4.1

Lettre	Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur
A	Wx1	-0,8228	Wx2	0,4911	Wx3	-0,3865	Ws1	-1,0000
Formule	Sx1	1,5441	Sx2	1,4502	Sx3	1,9512	Ws2	0,7500
(1-Q1) et Q2	Tetax1	0,0000	Tetax2	5,4845	Tetax3	0,0000	Ws3	0,7500
et Q3	Wy1	1,2031	Wy2	0,1398	Wy3	1,9512		
	Sy1	0,3389	Sy2	1,7697	Sy3	0,3412		
	Tetay1	1,6786	Tetay2	0,0000	Tetay3	1,7320		
C	Wx	0,4078						
Formule	Sx	0,6606						
Q1	Tetax	0,0000						
	Wy	0,9547						
	Sy	0,4675						
	Tetay	7,8112						
E	Wx1	0,8624	Wx2	0,1822	Ws1	0,7500		
Formule	Sx1	0,7142	Sx2	1,6289	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	0,0000				
	Wy1	0,2539	Wy2	0,9264				
	Sy1	1,0112	Sy2	0,3307				
	Tetay1	8,2117	Tetay2	7,7091				
F	Wx	0,2093						
Formule	Sx	0,8173						

Q1	Tetax	0,0000						
	Wy	0,7204						
	Sy	0,2293						
	Tetay	3,7411						
J	Wx	-0,5975	Ws1	-1,0000				
Formule	Sx	1,8699	Ws2*	1,5000	*S2=1			
1-Q1	Tetax	0,0000						
	Wy	0,3220						
	Sy	1,5918						
	Tetay	4,8592						
L	Wx	0,5930						
Formule	Sx	0,0425						
Q1	Tetax	0,0000						
	Wy	0,2628						
	Sy	0,9937						
	Tetay	6,0054						
M	Wx1	0,2539	Wx2	0,9264	Ws1	0,7500		
Formule	Sx1	1,0112	Sx2	0,3307	Ws2	0,7500		
Q1 et Q2	Tetax1	8,2117	Tetax2	7,7091				
	Wy1	0,8624	Wy2	0,1822				
	Sy1	0,7142	Sy2	1,6289				
	Tetay1	0,0000	Tetay2	0,0000				
N	Wx1	-0,5565	Wx2	1,9358	Ws1	0,7500		
Formule	Sx1	1,0997	Sx2	0,0140	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	4,9691				
	Wy1	1,6200	Wy2	-0,4804				
	Sy1	0,2764	Sy2	1,0706				
	Tetay1	3,7790	Tetay2	0,0000				
T	Wx1	0,4400	Wx2	0,9381	Ws1	0,7500		
Formule	Sx1	1,0806	Sx2	0,4557	Ws2	0,7500		

Q1 et Q2	Tetax1	0,0000	Tetax2	4,7448				
	Wy1	0,8122	Wy2	0,0559				
	Sy1	0,8784	Sy2	1,9514				
	Tetay1	10,4986	Tetay2	0,0000				

Tableau A-2 : Paramètres des MLQs produisant les images de la figure 4.3

Lettre	Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur
A	Wx1	-0,7548	Wx2	0,4903	Wx3	-0,3958	Ws1	-1,0000
Formule	Sx1	1,6341	Sx2	1,4500	Sx3	1,8614	Ws2	0,7500
(1-Q1) et Q2	Tetax1	0,0000	Tetax2	5,4834	Tetax3	0,0000	Ws3	0,7500
et Q3	Wy1	1,0958	Wy2	0,1416	Wy3	1,9577		
	Sy1	0,1266	Sy2	1,7707	Sy3	0,3033		
	Tetay1	1,7071	Tetay2	0,0000	Tetay3	1,7022		
C	Wx	0,3644						
Formule	Sx	0,8141						
Q1	Tetax	0,0000						
	Wy	0,9468						
	Sy	0,4069						
	Tetay	7,6765						
E	Wx1	0,8735	Wx2	0,2031	Ws1	0,7500		
Formule	Sx1	0,7007	Sx2	1,5907	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	0,0000				
	Wy1	0,2436	Wy2	0,9546				
	Sy1	1,0184	Sy2	0,2878				
	Tetay1	8,2146	Tetay2	7,5957				
F	Wx	0,2002						
Formule	Sx	0,8352						
Q1	Tetax	0,0000						
	Wy	0,6973						
	Sy	0,2332						
	Tetay	3,7413						

J	Wx	-0,6659	Ws1	-1,0000				
Formule	Sx	1,8496	Ws2*	1,5000	*S2=1			
1-Q1	Tetax	0,0000						
	Wy	0,3588						
	Sy	1,6346						
	Tetay	5,0264						
L	Wx	0,5829						
Formule	Sx	0,0115						
Q1	Tetax	0,0000						
	Wy	0,3372						
	Sy	1,0342						
	Tetay	6,1019						
M	Wx1	0,2436	Wx2	0,9546	Ws1	0,7500		
Formule	Sx1	1,0184	Sx2	0,2878	Ws2	0,7500		
Q1 et Q2	Tetax1	8,2146	Tetax2	7,5957				
	Wy1	0,8735	Wy2	0,2031				
	Sy1	0,7007	Sy2	1,5907				
	Tetay1	0,0000	Tetay2	0,0000				
N	Wx1	-0,4966	Wx2	1,9306	Ws1	0,7500		
Formule	Sx1	0,9954	Sx2	0,0115	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	4,9772				
	Wy1	1,4911	Wy2	-0,4830				
	Sy1	0,2957	Sy2	1,0703				
	Tetay1	3,7359	Tetay2	0,0000				
T	Wx1	0,4400	Wx2	0,9381	Ws1	0,7500		
Formule	Sx1	1,0806	Sx2	0,4557	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	4,7448				
	Wy1	0,8122	Wy2	0,0559				
	Sy1	0,8784	Sy2	1,9514				
	Tetay1	10,4986	Tetay2	0,0000				

Tableau A-3 : Paramètres des MLQs produisant les images de la figure 5.2

Lettre	Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur	
A	Wx1	0,5486					
Formule	Sx1	0,3484					
Q1	Tetax1	7,6452					
	Wx2	0,1829					
	Sx2	1,3074					
	Tetax2	9,6786					
	Wy	0,3126					
	Sy	1,4122					
	Tetay	0,0000					
F	Wx1	0,4473					
Formule	Sx1	0,3223					
Q1	Tetax1	0,0000					
	Wx2	0,1402					
	Sx2	1,9493					
	Tetax2	1,2106					
	Wy	0,5100					
	Sy	0,2600					
	Tetay	13,5000					
H	Wx1,1	0,3461	Wy2,1	0,3461	Ws1	-1,0000	
Formule	Sx1,1	1,0259	Sy2,1	1,0259	Ws2	1,5000	
(1-Q1)	Tetax1,1	0,0000	Tetay2,1	0,0000	Tetas2	10,0000	
ou Q2	Wx1,2	0,7434	Wy2,2	0,7434	Ws3*	1,5000	*S3=1
	Sx1,2	0,4082	Sy2,2	0,4082			
	Tetax1,2	7,1942	Tetay2,2	7,1942			
I	Wy1	0,5943	Ws1	-1,0000			
Formule	Sy1	0,7196	Ws2*	1,5000	*S2=1		
1-Q1	Tetay1	0,0000					
	Wy2	0,8572					

	Sy2	0,3348					
	Tetay2	6,1225					
O	Wx1	0,3461					
Formule	Sx1	1,0259					
Q1	Tetax1	0,0000					
	Wx2	0,7434					
	Sx2	0,4082					
	Tetax2	7,1942					
	Wy1	0,3461					
	Sy1	1,0259					
	Tetay1	20,0000					
	Wy2	0,7434					
	Sy2	0,4082					
	Tetay2	27,1942					
U	Wx1	0,9900	Wy2,1	0,3461	Ws1	-1,0000	
Formule	Sx1	0,7200	Sy2,1	1,0259	Ws2	1,5000	
(1-Q1)	Tetax1	0,0000	Tetay2,1	0,0000	Tetas2	10,0000	
ou Q2			Wy2,2	0,7434	Ws3*	1,5000	*S3=1
			Sy2,2	0,4082			
			Tetay2,2	7,1942			

Tableau A-4 : Paramètres des MLQs produisant les images de la figure 12.4

Lettre	Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur	Paramètre	Valeur
A	Wx1	-0,0648	Wx2	0,4903	Wx3	-0,3958	Ws1	-1,0000
Formule	Sx1	1,8841	Sx2	1,4500	Sx3	1,8614	Ws2	0,7500
(1-Q1)	Tetax1	0,0000	Tetax2	6,4834	Tetax3	0,0000	Ws3	0,7500
et Q2 et Q3	Wy1	1,0808	Wy2	0,1666	Wy3	1,9577		
	Sy1	0,1266	Sy2	1,7707	Sy3	0,3333		
	Tetay1	1,5071	Tetay2	0,0000	Tetay3	1,8022		
B	Wx1,1	0,9999	Wy2,1	0,9999	Wx3	0,9999	Ws1	0,4000
Formule	Sx1,1	0,2250	Sy2,1	0,2250	Sx3	0,0250	Ws2	0,4000

Q1 et Q2	Tetax1,1	0,0000	Tetay2,1	0,0000	Tetax3	0,0000	Ws3	0,4000
et Q3	Wx1,2	0,1000	Wy2,2	0,1000	Wy3	0,9999		
	Sx1,2	1,0000	Sy2,2	1,0000	Sy3	0,8250		
	Tetax1,2	9,6750	Tetay2,2	15,4750	Tetay3	3,7375		
	Wy1	0,9999	Wx2	0,9999				
	Sy1	0,2250	Sx2	0,6250				
	Tetay1	3,9375	Tetax2	7,5375				
C	Wx	0,3644						
Formule	Sx	0,8141						
Q1	Tetax	0,0000						
	Wy	0,9468						
	Sy	0,4069						
	Tetay	7,6765						
D	Wx1,1	0,9999	Wy2,1	0,9999	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,2250	Sy2,1	0,2250	Sx3	0,0250	Sx4	0,8250
Q1 et Q2	Tetax1,1	0,0000	Tetay2,1	0,0000	Tetax3	0,0000	Tetax4	0,0000
et Q3 et Q4	Wx1,2	0,1000	Wy2,2	0,1000	Wy3	0,9999	Wy4	0,9999
	Sx1,2	1,0000	Sy2,2	1,0000	Sy3	0,8250	Sy4	0,8250
	Tetax1,2	13,2750	Tetay2,2	13,4750	Tetay3	1,9375	Tetay4	9,9375
	Wy1	0,9999	Wx2	0,9999				
	Sy1	0,2250	Sx2	0,4250				
	Tetay1	7,5375	Tetax2	7,5375				
	Ws1	0,3000						
	Ws2	0,3000						
	Ws3	0,3000						
	Ws4	0,3000						
E	Wx1	0,8735	Wx2	0,2031	Ws1	0,7500		
Formule	Sx1	0,6707	Sx2	1,1907	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	0,0000				
	Wy1	0,2436	Wy2	0,9546				
	Sy1	1,0184	Sy2	0,2278				
	Tetay1	7,6146	Tetay2	5,5957				

F	Wx	0,2502						
Formule	Sx	0,8352						
Q1	Tetax	0,0000						
	Wy	0,6723						
	Sy	0,2782						
	Tetay	4,6413						
G	Wx1,1	0,9999	Wy2,1	0,9999	Wx3	0,9999	Ws1	0,4000
Formule	Sx1,1	0,2250	Sy2,1	0,2250	Sx3	0,2250	Ws2	0,4000
Q1 et Q2	Tetax1,1	0,0000	Tetay2,1	0,0000	Tetax3	0,0000	Ws3	0,4000
et Q3	Wx1,2	0,1000	Wy2,2	0,1000	Wy3	0,9999		
	Sx1,2	1,0000	Sy2,2	1,0000	Sy3	0,2250		
	Tetax1,2	13,2750	Tetay2,2	15,4750	Tetay3	3,9375		
	Wy1	0,9999	Wx2	0,9999				
	Sy1	0,2250	Sx2	0,6250				
	Tetay1	7,5375	Tetax2	7,5375				
H	Wx1,1	0,3461	Wy2,1	0,3461	Ws1	-1,0000		
Formule	Sx1,1	1,0259	Sy2,1	1,0259	Ws2	1,5000		
(1-Q1) ou Q2	Tetax1,1	0,0000	Tetay2,1	0,0000	Tetas2	10,0000		
	Wx1,2	0,7434	Wy2,2	0,7434	Ws3*	1,5000	*S3=1	
	Sx1,2	0,4082	Sy2,2	0,4082				
	Tetax1,2	7,1942	Tetay2,2	7,1942				
I	Wy1	0,5443	Ws1	-1,0000				
Formule	Sy1	0,7196	Ws2*	1,5000	*S2=1			
1-Q1	Tetay1	0,0000						
	Wy2	0,8572						
	Sy2	0,3348						
	Tetay2	6,6225						
J	Wx	-0,6659	Ws1	-1,0000				
Formule	Sx	2,4496	Ws2*	1,5000				
1-Q1	Tetax	0,0000						
	Wy	0,4188						
	Sy	1,6346						
	Tetay	5,0264						

			*S2=1					
K	Wx1,1	0,9999	Wx2,1	0,9990	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,0250	Sx2,1	0,0250	Sx3	0,6250	Sx4	0,8250
Q1 et Q2	Tetax1,1	0,0000	Tetax2,1	0,0000	Tetax3	5,7375	Tetax4	7,5375
et Q3 et Q4	Wx1,2	0,1000	Wx2,2	0,1000	Wy3	0,9999	Wy4	0,9999
et Q5	Sx1,2	1,0000	Sx2,2	1,0000	Sy3	0,2250	Sy4	0,2250
	Tetax1,2	9,4750	Tetax2,2	9,4750	Tetay3	0,0000	Tetay4	0,0000
	Wy1	0,9999	Wy2	0,9999				
	Sy1	0,2250	Sy2	0,2250				
	Tetay1	1,9375	Tetay2	3,7350				
	Wx5	0,9999	Ws1	0,2200				
	Sx5	0,4250	Ws2	0,2200				
	Tetax5	0,0000	Ws3	0,2200				
	Wy5	0,9999	Ws4	0,2200				
	Sy5	0,8250	Ws5	0,2200				
	Tetay5	5,9375						
L	Wx	0,7329						
Formule	Sx	0,0415						
Q1	Tetax	0,0000						
	Wy	0,2872						
	Sy	1,0342						
	Tetay	6,1019						
M	Wx1	0,2436	Wx2	0,9546	Ws1	0,7500		
Formule	Sx1	1,0184	Sx2	0,2278	Ws2	0,7500		
Q1 et Q2	Tetax1	7,6146	Tetax2	5,5957				
	Wy1	0,8735	Wy2	0,2031				
	Sy1	0,6707	Sy2	1,1907				
	Tetay1	0,0000	Tetay2	0,0000				
N	Wx1	-0,5565	Wx2	1,6498	Ws1	0,7500		
Formule	Sx1	1,0997	Sx2	0,0440	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	3,6910				
	Wy1	1,6080	Wy2	-0,4804				
	Sy1	0,3364	Sy2	1,0706				

	Tetay1	3,3790	Tetay2	0,0000				
O	Wx1,1	0,9999						
Formule	Sx1,1	0,3250						
Q1	Tetax1,1	0,0000						
	Wx1,2	0,1000						
	Sx1,2	1,0000						
	Tetax1,2	13,3000						
	Wy	0,9999						
	Sy	0,3250						
	Tetay	6,6500						
P	Wx1	0,5486						
Formule	Sx1	0,3484						
Q1	Tetax1	7,6452						
	Wx2	0,1829						
	Sx2	1,3074						
	Tetax2	9,6786						
	Wy	0,3126						
	Sy	1,0922						
	Tetay	0,0000						
Q	Wx1,1	0,9999	Wx2,1	0,9999	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,2250	Sx2,1	0,2250	Sx3	0,0250	Sx4	0,9250
Q1 et Q2	Tetax1,1	0,0000	Tetax2,1	0,0000	Tetax3	0,0000	Tetax4	7,3375
et Q3 et Q4	Wx1,2	0,1000	Wx2,2	0,1000	Wy3	0,9999	Wy4	0,9999
	Sx1,2	1,0000	Sx2,2	1,0000	Sy3	0,9250	Sy4	0,0250
	Tetax1,2	11,9250	Tetax2,2	11,9250	Tetay3	7,3375	Tetay4	0,0000
	Wy1	0,9999	Wy2	0,9999				
	Sy1	0,2250	Sy2	0,2250				
	Tetay1	6,6375	Tetay2	5,2875				
	Ws1	0,3000						
	Ws2	0,3000						
	Ws3	0,3000						
	Ws4	0,3000						

R	Wx1,1	0,9999	Wx2	0,9999	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,2250	Sx2	0,6250	Sx3	0,8250	Sx4	0,0250
Q1 et Q2	Tetax1,1	0,0000	Tetax2	5,7375	Tetax3	7,5375	Tetax4	0,0000
et Q3 et Q4	Wx1,2	0,1000	Wy2	0,9999	Wy3	0,9999	Wy4	0,9999
	Sx1,2	1,0000	Sy2	0,2250	Sy3	0,2250	Sy4	0,8250
	Tetax1,2	9,6750	Tetay2	0,0000	Tetay3	0,0000	Tetay4	5,5375
	Wy1	0,9999						
	Sy1	0,2250						
	Tetay1	3,9375						
	Ws1	0,3000						
	Ws2	0,3000						
	Ws3	0,3000						
	Ws4	0,3000						
S	Wx1	0,9999	Wy2,1	0,9999	Ws1	0,7500		
Formule	Sx1	0,2250	Sy2,1	0,0250	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetay2,1	0,0000				
	Wy1	0,9999	Wy2,2	0,1000				
	Sy1	0,2250	Sy2,2	1,0000				
	Tetay1	3,9375	Tetay2,2	15,2750				
			Wx2	0,9999				
			Sx2	0,6250				
			Tetax2	7,3375				
T	Wx1	0,2900	Wx2	0,9381	Ws1	0,7500		
Formule	Sx1	1,0806	Sx2	0,3557	Ws2	0,7500		
Q1 et Q2	Tetax1	0,0000	Tetax2	4,7448				
	Wy1	0,6922	Wy2	0,0559				
	Sy1	0,8784	Sy2	1,3514				
	Tetay1	10,4986	Tetay2	0,0000				
U	Wx1	0,9999						
Formule	Sx1	0,0250						
Q1	Tetax1	0,0000						
	Wx2	0,1000						
	Sx2	1,0000						

	Tetax2	13,1750						
	Wy	0,9999						
	Sy	0,3250						
	Tetay	6,4375						
V	Wx1,1	0,9999	Wx2,1	0,9999	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,0250	Sx2,1	0,0250	Sx3	0,8250	Sx4	0,8250
Q1 et Q2	Tetax1,1	0,0000	Tetax2,1	0,0000	Tetax3	1,9375	Tetax4	0,0000
et Q3 et Q4	Wx1,2	0,1000	Wx2,2	0,1000	Wy3	0,9999	Wy4	0,9999
	Sx1,2	1,0000	Sx2,2	1,0000	Sy3	0,0250	Sy4	0,8250
	Tetax1,2	13,0750	Tetax2,2	13,2750	Tetay3	0,0000	Tetay4	9,9375
	Wy1	0,9999	Wy2	0,9999				
	Sy1	0,2250	Sy2	0,4250				
	Tetay1	5,5375	Tetay2	7,3375				
	Ws1	0,3000						
	Ws2	0,3000						
	Ws3	0,3000						
	Ws4	0,3000						
W	Wx1,1	0,9999	Wx2,1	0,9999	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,0250	Sx2,1	0,0250	Sx3	0,8250	Sx4	0,8250
Q1 et Q2	Tetax1,1	0,0000	Tetax2,1	0,0000	Tetax3	1,9375	Tetax4	5,9375
et Q3 et Q4	Wx1,2	0,1000	Wx2,2	0,1000	Wy3	0,9999	Wy4	0,9999
et Q5 et Q6	Sx1,2	1,0000	Sx2,2	1,0000	Sy3	0,0250	Sy4	0,4250
	Tetax1,2	9,4750	Tetax2,2	9,4750	Tetay3	0,0000	Tetay4	0,0000
	Wy1	0,9999	Wy2	0,9999				
	Sy1	0,2250	Sy2	0,2250				
	Tetay1	1,9375	Tetay2	5,5375				
	Wx5,1	0,9999	Wx6	0,9999	Ws1	0,1800		
	Sx5,1	0,0250	Sx6	0,8250	Ws2	0,1800		
	Tetax5,1	0,0000	Tetax6	0,0000	Ws3	0,1800		
	Wx5,2	0,1000	Wy6	0,9999	Ws4	0,1800		
	Sx5,2	1,0000	Sy6	0,8250	Ws5	0,1800		
	Tetax5,2	13,4750	Tetay6	9,9375	Ws6	0,1800		
	Wy5	0,9999						

	Sy5	0,6250						
	Tetay5	5,5375						
X	Wx1,1	0,9999	Wx2,1	0,9999	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,0250	Sx2,1	0,0250	Sx3	0,2750	Sx4	0,4250
Q1 et Q2	Tetax1,1	0,0000	Tetax2,1	0,0000	Tetax3	0,0000	Tetax4	0,0000
et Q3 et Q4	Wx1,2	0,1000	Wx2,2	0,1000	Wy3	0,9999	Wy4	0,9999
et Q5 et Q6	Sx1,2	1,0000	Sx2,2	1,0000	Sy3	0,8750	Sy4	0,7250
et Q7 et Q8	Tetax1,2	8,6250	Tetax2,2	8,7750	Tetay3	7,1375	Tetay4	5,9375
	Wy1	0,9999	Wy2	0,9999				
	Sy1	0,2750	Sy2	0,4250				
	Tetay1	1,4875	Tetay2	2,8375				
	Wx5	0,9999	Wx6	0,9999	Wy7,1	0,9999	Wy8,1	0,9999
	Sx5	0,8750	Sx6	0,7250	Sy7,1	0,0250	Sy8,1	0,0250
	Tetax5	7,1375	Tetax6	5,9375	Tetay7,1	0,0000	Tetay8,1	0,0000
	Wy5	0,9999	Wy6	0,9999	Wy7,2	0,1000	Wy8,2	0,1000
	Sy5	0,2750	Sy6	0,4250	Sy7,2	1,0000	Sy8,2	1,0000
	Tetay5	0,0000	Tetay6	0,0000	Tetay7,2	8,6250	Tetay8,2	8,7750
					Wx7	0,9999	Wx8	0,9999
					Sx7	0,2750	Sx8	0,4250
					Tetax7	1,4875	Tetax8	2,8375
	Ws1	0,1400						
	Ws2	0,1400						
	Ws3	0,1400						
	Ws4	0,1400						
	Ws5	0,1400						
	Ws6	0,1400						
	Ws7	0,1400						
	Ws8	0,1400						
Y	Wx1,1	0,9999	Wx2,1	0,9999	Wx3	0,9999	Wx4	0,9999
Formule	Sx1,1	0,0250	Sx2,1	0,0250	Sx3	0,4250	Sx4	0,6250
Q1 et Q2	Tetax1,1	0,0000	Tetax2,1	0,0000	Tetax3	1,9375	Tetax4	3,7375
et Q3 et Q4	Wx1,2	0,1000	Wx2,2	0,1000	Wy3	0,9999	Wy4	0,9999
et Q5 et Q6	Sx1,2	1,0000	Sx2,2	1,0000	Sy3	0,0250	Sy4	0,0250

	Tetax1,2	9,4750	Tetax2,2	9,6750	Tetay3	0,0000	Tetay4	0,0000
	Wy1	0,9999	Wy2	0,9999				
	Sy1	0,2250	Sy2	0,4250				
	Tetay1	1,9375	Tetay2	3,7375				
	Wx5	0,9999	Wx6	0,9999	Ws1	0,1800		
	Sx5	0,6250	Sx6	0,4250	Ws2	0,1800		
	Tetax5	0,0000	Tetax6	0,0000	Ws3	0,1800		
	Wy5	0,9999	Wy6	0,9999	Ws4	0,1800		
	Sy5	0,6250	Sy6	0,8250	Ws5	0,1800		
	Tetay5	9,7375	Tetay6	9,5375	Ws6	0,1800		
Z	Wy1,1	0,9999	Wy2,1	0,9999	Wy3,1	0,9999	Wx4	0,9999
Formule	Sy1,1	0,0250	Sy2,1	0,0250	Sy3,1	0,0250	Sx4	0,6250
Q1 et Q2	Tetay1,1	0,0000	Tetay2,1	0,0000	Tetay3,1	0,0000	Tetax4	0,0000
et Q3 et Q4	Wy1,2	0,1000	Wy2,2	0,1000	Wy3,2	0,1000	Wy4	0,9999
et Q5 et Q6	Sy1,2	1,0000	Sy2,2	1,0000	Sy3,2	1,0000	Sy4	0,5250
	Tetay1,2	8,5757	Tetay2,2	8,5750	Tetay3,2	8,5750	Tetay4	7,9375
	Wx1	0,9999	Wx2	0,9999	Wx3	0,9999		
	Sx1	0,2250	Sx2	0,2250	Sx3	0,2250		
	Tetax1	4,6375	Tetax2	2,8375	Tetax3	1,0375		
	Wx5	0,9999	Wx6	0,9999	Ws1	0,1800		
	Sx5	0,4250	Sx6	0,2250	Ws2	0,1800		
	Tetax5	0,0000	Tetax6	0,0000	Ws3	0,1800		
	Wy5	0,9999	Wy6	0,9999	Ws4	0,1800		
	Sy5	0,7250	Sy6	0,9250	Ws5	0,1800		
	Tetay5	7,7375	Tetay6	7,5375	Ws6	0,1800		

ANNEXE 2 – Comparaisons

Se trouve ici un tableau comparant le nombre d'éléments et le nombre de paramètres des réseaux MLQ qui produisent les lettres finales à ceux de leurs équivalents MLP produisant des lettres semblables. Les lettres suivies d'un astérisque sont celles produites par la méthode de construction. Ce tableau fait particulièrement ressortir l'avantage que la puissance du quantron apporte sur la diminution de la complexité des réseaux et le désavantage des trois paramètres nécessaires pour chaque entrée.

Tableau A-5 : Tableau de comparaison avec le MLP

Lettre produite	MLQ		MLP	
	# quantrons	# paramètres	# perceptrons	# paramètres
A	4	27	10	27
B*	4	33	12	33
C	1	6	4	10
D*	5	42	11	27
E	3	18	7	18
F	1	6	6	15
G*	4	33	10	24
H	3	21	6	15
I	2	12	3	7
J	2	12	8	26
K*	6	51	13	32
L	1	6	5	12
M	3	18	7	18
N	3	12	6	17
O*	1	9	5	13
P	1	9	8	21
Q*	5	42	15	39
R*	5	69	12	29
S*	3	21	10	26
T	3	12	5	12
U*	1	9	4	10
V*	5	14	7	21
W*	7	63	17	55
X*	9	84	7	21
Y*	7	60	10	32
Z*	7	63	6	17
Lettres alléatoires	27	159	75	198
Lettres construites	69	593	139	379
Somme des lettres	96	752	214	577

ANNEXE 3 – Codes Matlab

Se trouvent ici les codes essentiels à l'implémentation du quantron sur le logiciel Matlab.

```
%FICHIER "quantron.m"
%Fonction principale pour le calcul du quantron
function [B,T]=quantron(X,W,S,Teta,Gamma,N,Noyau)

    nbpts=1000; %Précision de l'approximation
    nbin=length(X);

    for k=1:nbin
        tk(k)=Teta(k)+(N(k)-1)*X(k)+2*S(k);
    end
    tmax=max(tk);
    t=linspace(0,tmax,nbpts);

    consum=0;
    for k=1:nbin
        if X(k)==0
            eval(['con',num2str(k),'=zeros(1,nbpts);']);
        else

            eval(['con',num2str(k),'=conv(peigne(t,Teta(k),X(k),N(k)),',Noyau,
            '(t,S(k)))*W(k);']);
            eval(['consum=consum+con',num2str(k),'(1:length(t));']);
        end
    end
    seuilhits=find(consum>=Gamma);
    hit=sum(seuilhits);

    if hit>0
        B=1;
        T=t(seuilhits(1));
    else
        B=0;
        T=0;
    end
end

%FICHIER "peigne.m"
%Création d'un peigne de Dirac pour la convolution
function y=peigne(t,teta,x,n)
   res=t(2)-t(1);
   y=zeros(1,length(t));
   for k=0:n-1
       if ceil(teta/res+k*x/res)<=length(t)&&ceil(teta/res+k*x/res)~=0
          y(ceil(teta/res+k*x/res))=1;
       end
```

```
        if ceil(teta/res+k*x/res)==0
            y(1)=1;
        end
    end
end

%FICHIER "carre.m"
%Formation du noyau de type carré
function y=carre(t,s)
    x=heaviside(t)-heaviside(t-s);
    x(isnan(x))=1;
    step=t(2)-t(1);
    long=ceil(s/step)+1;
    y=x(1:long);
end



%FICHIER "triangle.m"
%Formation du noyau de type triangle
function y=triangle(t,s)
    x=(1/s)*(t.*heaviside(t)+(2*s-2*t).*heaviside(t-s)+(t-2*s).*heaviside(t-
     2*s));
    x(isnan(x))=0;

    step=t(2)-t(1);
    long=ceil(2*s/step)+1;
    y=x(1:long);
end



%FICHIER "parabolle.m"
%Formation du noyau de type parabole
function y=parabolle(t,s)
    x=(1/s^2)*((heaviside(t)-heaviside(t-2*s)).*(-t.^2+2*s*t));
    x(isnan(x))=0;
    step=t(2)-t(1);
    long=ceil(2*s/step)+1;
    y=x(1:long);
end



%FICHIER "vague.m"
%Formation du noyau de type vague
function y=vague(t,s)
    a=1.25;
    V=@(u,v) (1-normcdf(log(a)./real(sqrt(u)+10^-5))).*(1-
     normcdf(log(a)./(sqrt(v)+10^-5)));
```

```
    x=V(t,s)+(1-V(t,s)-V(t-s,s)).*heaviside(t-s)+(V(t-s,s)-1).*heaviside(t-
     2*s);
    x(isnan(x))=0;
    step=t(2)-t(1);
    long=ceil(2*s/step)+1;
    y=x(1:long);
end
```