

Προσαρμοστική Μηχανική Μάθηση

Hierarchical Temporal Memory και νευρομορφικό hardware

Κωνσταντίνος Σαμαράς-Τσακίρης

January 30, 2017

Παρατήρηση

Καθώς το κείμενο αυτό είναι συμπληρωματικό της παρουσίασης, θα γίνονται αναφορές σε αυτή και σε άλλο υλικό όπου αυτό κρίνεται επαρκές. Γενικά προτείνεται να διαβαστεί παράλληλα με την παρουσίαση.

1 Hierarchical Temporal Memory

1.1 Το πρόβλημα της πρόβλεψης ακολουθιών

Όπως μας πληροφορεί ο LeCun [8], τα σύνορα της τεχνητής νοημοσύνης ξεπερνούν το πρόβλημα της εκμάθησης ενός στατικού συνόλου δεδομένων με supervised μεθόδους, όπως γίνεται σε πολλές σύγχρονες εφαρμογές. Η τεχνητή νοημοσύνη καλείται σήμερα να αξιοποιήσει τις τεράστιες και συνεχείς ροές δεδομένων που παρέχει το (ανθρωπογενές και μη) περιβάλλον, σε πραγματικό χρόνο, και γενικώς δίχως την πολυτέλεια της προσήμανσης που απαιτεί η επιβλεπόμενη μάθηση.

Επομένως, καλούμαστε να μοντελοποιήσουμε έναν κόσμο που αλλάζει [14]. Το μοντέλο οφείλει ή να είναι γενικότερο από όλες τις δυνατές μεταβολές του κόσμου ή να αλλάζει μαζί του. Η έννοια της ροής δεδομένων που αναφέρθηκε υπονοεί την έννοια του χρόνου, της αλληλουχίας, που δίνει στο μοντέλο μας ένα νέο εργαλείο: την αιτιώδη σχέση. Την ερμηνεία του κόσμου με τη βοήθεια της συνεπαγωγής και της αναγνώρισης απαιτήσεων και συνεπειών.

Θεσμοθετείται έτσι το πρόβλημα της εκμάθησης ακολουθιών (sequence learning). Δηλαδή: ο πράκτορας, που παρακολουθεί μια αλληλουχία δεδομένων (γεγονότων), καλείται να *προβλέψει* τη συνέχεια και να *δράσει* έτσι, ώστε να ανταμειφθεί. Η ανταμοιβή, το κίνητρο και γενικότερα ο στόχος, έχει τεθεί από εξωτερικό παράγοντα και βρίσκεται εκτός του πλαισίου της νοημοσύνης. Η παρατήρηση αυτή προσδιορίζει σε κάποιο βαθμό τι εννοείται με τον όρο “νοημοσύνη”, η οποία δεν αρκεί για τη λειτουργία ενός ζωντανού οργανισμού. Μια άμεση εφαρμογή του προβλήματος της πρόβλεψης είναι και η *αναγνώριση ανωμαλιών*.

Η εκμάθηση ακολουθιών είναι κλασικό πρόβλημα, για το οποίο έχουν αναπτυχθεί κλασικές λύσεις. Κυριότερο και ευρύτερα διαδεδομένο είδος μοντέλων, ειδικά πριν τη σύγχρονη επάνοδο των νευρωνικών δικτύων, είναι τα Hidden Markov Models. Τα κλασικά (στυλ perceptron) νευρωνικά δίκτυα προσφέρουν στο πρόβλημα τα “time-delay neural networks” (TDNN). Η ουσιαστική συνεισφορά των κλασικών νευρωνικών δικτύων επιτυγχάνεται όμως με τα ανάδρομα δίκτυα (recurrent neural networks), που γενίκευσαν τις παλαιότερες feedforward αρχιτεκτονικές ακριβώς για να μπορούν να αντιμετωπίσουν εγγενώς προβλήματα ακολουθιών. Ιδιαίτερης μνείας χρήζει η δομή “Long short-term memory” (LSTM). Όλα όμως αυτά τα μοντέλα έχουν σημαντικούς περιορισμούς.

Ο ανθρώπινος εγκέφαλος, συγκεκριμένα ο φλοιός (neocortex), είναι το βασικότερο παράδειγμα ευφυούς συστήματος που επιλύει το παραπάνω πρόβλημα. Θα ήταν εύλογο λοιπόν να μοντελοποιηθεί η λειτουργία του με αλγοριθμικό τρόπο, που θα στηρίζεται στις βασικές αρχές της εγκεφαλικής λειτουργίας. Για ένα τέτοιο εγχείρημα βεβαίως απαιτείται να γίνει κατανοητό ποια χαρακτηριστικά του εγκεφάλου όντως απαιτούνται για τη λειτουργία του αυτή.

Ίσως φανούν χρήσιμες μερικές νευρολογικές παρατηρήσεις για τον εγκεφαλικό φλοιό:

- Η μορφή του είναι διδιάστατη, με τους νευρώνες να ταξιθετούνται σε διακριτά, στοιβαγμένα επίπεδα.
- Η δομή του είναι ομοιόμορφη σε όλη του την έκταση. Μπορεί να θεωρηθεί ότι κάθε περιοχή του neocortex λύνει το ίδιο πρόβλημα με τον ίδιο τρόπο.

- Οι περιοχές ενώνονται ιεραρχικά. Επομένως, οι ιεραρχικά ανώτερες περιοχές υλοποιούν πιο αφηρημένες ιδέες.
- Οι νευρώνες ενεργοποιούνται σποραδικά και πολύ σπάνια. Αν σχηματίσουμε ένα διάνυσμα με την κατάσταση κάθε νευρώνα, το διάνυσμα αυτό θα είναι αραιό (οι περισσότερες θέσεις 0, ανενεργές)

1.2 Sparse Distributed Representations

Με βάση την παρατήρηση της αραιής πυροδότησης των νευρώνων στον εγκέφαλο σχεδιάζουμε ένα μεγάλο, αραιό, δυαδικό διάνυσμα που αναπαριστά την πυροδότηση ή μη κάθε νευρώνα της μελετούμενης περιοχής. Μπορεί να θεωρηθεί ότι κάθε περιοχή ανταλλάσσει τέτοια μεγάλα αραιά δυαδικά διανύσματα με τις επόμενες. Ουσιαστικά, αυτά τα αραιά διανύσματα αποτελούν τη “δομή δεδομένων” [4, 6] του εγκεφάλου. Το ιδιαίτερο χαρακτηριστικό αυτών των διανυσμάτων, λόγω αυτού που αναπαριστούν, είναι ότι κάθε ξεχωριστό bit τους φέρει σημασιολογικό νόημα – αντίθετα δηλαδή από την κωδικοποίηση ενός γράμματος σε κώδικα ASCII, όπου νόημα φέρει μόνο η πλήρης “λέξη”. Εξού και το “distributed” του ονόματος.

1.2.1 Χωρητικότητα SDR

Η χωρητικότητα ενός SDR με μέγεθος n και cardinality (πλήθος μονάδων) w είναι το πλήθος των διαφορετικών SDR με αυτή τη μορφή, δηλαδή οι συνδυασμοί των n ανά w :

$$\binom{n}{w} = \frac{n!}{w!(n-w)!}$$

1.2.2 Συνδυασμοί SDR

Διαφάνεια 9.

1.2.3 Σύνολα SDR

Διαφάνεια 10.

Επισημαίνεται μονάχα η *σχεδιαστική ελευθερία* που παρέχουν τα SDR. Όσο μεγαλύτερα SDR χρησιμοποιούνται, τόσο αυξάνεται ο χώρος των ενδεχομένων και τόσο πιο απίθανα γίνονται τα λανθασμένα ταιριάσματα.

1.3 Hierarchical Temporal Memory (Numenta) Overview

Αξιοποιώντας τις προηγούμενες παρατηρήσεις για τον εγκεφαλικό φλοιό και τις θεμελιώδεις μαθηματικές ιδιότητες των Sparse Distributed Representations μπορεί να μοντελοποιηθεί η λειτουργία του με ένα υπολογιστικό σύστημα. Το βασικό χαρακτηριστικό του συστήματος αυτού είναι η κωδικοποίηση και ανάκληση ακολουθιών, ενώ δομείται ιεραρχικά όπως ο εγκεφαλικός φλοιός, και γι’ αυτό καλείται Hierarchical Temporal Memory (HTM). Εδώ βέβαια ο σκοπός του HTM δεν είναι να μοντελοποιήσει τον εγκεφαλικό φλοιό, αλλά με βάση αυτόν να αντιμετωπίσει το πρόβλημα της εκμάθησης ακολουθιών. Όλο το σύστημα που περιγράφεται έχει σχεδιαστεί από την εταιρία Numenta [4].

Ένα σύστημα HTM λοιπόν αποτελεί ένα τεχνητό νευρωνικό δίκτυο, αλλά βασιζόμενο σε *έννοιες πολύ διαφορετικές* από τα μοντέλα που συνήθως καλούνται τεχνητά νευρωνικά δίκτυα. Παρακάτω αναλύεται η βασική του δομή.

1.3.1 Μοντέλο νευρώνα

Πρώτα απ’ όλα: νευρώνα ονομάζουμε μια δομή που δέχεται σήματα από άλλους νευρώνες και, όταν αυτά πληρούν κάποιες προϋποθέσεις, ενεργοποιείται, για να στείλει σήματα στους επόμενους νευρώνες. Η επαφή 2 νευρώνων καλείται σύναψη.

Ο βιολογικός νευρώνας (συγκεκριμένα πυραμιδοειδής νευρώνας που απαντά συχνά στο neocortex) είναι ένα σύνθετο στοιχείο, με πολλούς χώρους και διανεμημένες λειτουργίες, που υλοποιούν τις λογικές πράξεις της άθροισης, του πολλαπλασιασμού, της χωρικής και χρονικής ολοκλήρωσης ταυτόχρονα και παράλληλα σε διαφορετικές ομάδες εισόδων. Η κεντρική δομή του νευρώνα, το σώμα, δέχεται ερεθίσματα από το προηγούμενο ιεραρχικά επίπεδο. Αν έρθουν αρκετά τέτοια ερεθίσματα σε σύντομο χρονικό διάστημα ο

νευρώνας ενεργοποιείται. Οι πηγές των σημάτων που ο νευρώνας λαμβάνει στο σώμα του ονομάζονται *receptive field* του νευρώνα. Περιφερειακά του σώματος υπάρχουν τα δενδριτικά τμήματα, στα οποία υπάρχουν συνάψεις με νευρώνες του ίδιου επιπέδου, που παρέχουν την πληροφορία των συμφραζόμενων. Κάθε τμήμα αυτόνομα δέχεται σήματα και, αν αυτά είναι αρκετά, προκαλείται ένα NMDA spike. Σε αντίθεση με την ενεργοποίηση του σώματος, αυτό γενικά δεν αρκεί για να ενεργοποιήσει ολόκληρο το νευρώνα· θέτει όμως το νευρώνα σε κατάσταση “επιφυλακής”, διευκολύνοντας τη μετέπειτα ενεργοποίησή του από σωματικά ερεθίσματα. Ο νευρώνας επίσης λαμβάνει ερεθίσματα από την πλευρά του άξονα, από τους νευρώνες δηλαδή που “δέχονται” το σήμα του στην κανονική του πορεία. Αυτά αποτελούν σήματα ανάδρασης.

Μια απλοποιημένη εκδοχή αυτής της περιγραφής αποτελεί και ο υπολογιστικός νευρώνας του συστήματος που επιδεικνύεται. Έχει πάλι 3 περιοχές που συμπεριφέρονται διαφορετικά για να λαμβάνει εισόδους feedforward, context και feedback όπως ο βιολογικός. Επαρκής feedforward είσοδος αρκεί για να τον ενεργοποιήσει, ενώ επαρκής είσοδος context ή feedback τον θέτει στο λεγόμενο “predictive state”. Αν στη συνέχεια (στη συνέχεια... ακολουθία... συνεπαγωγή...) έρθει feedforward είσοδος, θα ενεργοποιηθεί πιο εύκολα και γρήγορα απ’ό,τι άλλοι νευρώνες με την ίδια είσοδο. Σε αντιπαράβολή, ο “νευρώνας” ενός κλασικού τεχνητού νευρωνικού δικτύου δεν έχει διακριτά λειτουργικά τμήματα και απλώς αθροίζει όλες τις εισόδους του.

1.3.2 Μοντέλο δικτύου

Και μόνο ο παραπάνω ανάδρομος ορισμός του νευρώνα παραπέμπει στο ρόλο του ως μονάδα δικτύου.

Η πρώτη οργανωτική δομή νευρώνων, τόσο στο neocortex, όσο και στο HTM, είναι το minicolumn. Συνοπτικά, ο ρόλος των minicolumns είναι να επιτρέπουν διαφορετικές εσωτερικές αναπαραστάσεις των εξωτερικών (feedforward) ερεθισμάτων, ανάλογα με τα συμφραζόμενα. Όλοι οι νευρώνες του minicolumn μοιράζονται *το ίδιο receptive field*, ενώ μεταξύ τους υπάρχει *inhibition*. Έτσι υλοποιείται μια αρχιτεκτονική “winner-takes-all”, όπου, αν κάποιοι νευρώνες είναι ήδη σε predictive κατάσταση όταν έρθει feedforward είσοδος που ερεθίζει το minicolumn, θα ενεργοποιηθούν πρώτοι και θα αποτρέψουν την ενεργοποίηση των υπολοίπων. Η παραπάνω περιγραφή βέβαια προτρέπει λίγο.

Πολλά minicolumns συνθέτουν ένα cortical layer, που αποτελεί τη λειτουργική μονάδα ενός συστήματος HTM. Κάθε minicolumn του cortical layer έχει διαφορετικές feedforward συνδέσεις προς την είσοδο του layer, ενώ διαφορετικά minicolumns μοιράζονται μεταξύ τους συνδέσεις που φέρουν πληροφορίες συμφραζόμενων. Σε πολλές περιπτώσεις μπορεί να υπάρχει ολικό inhibition μεταξύ των minicolumns. Αν βέβαια το cortical layer έχει κάποια τοπολογική διάταξη προς την είσοδό του¹, όπως συχνά συμβαίνει στον εγκεφαλικό φλοιό, τότε οι inhibitory συνδέσεις είναι τοπικές και ένα minicolumn δεν ανταγωνίζεται κάποιο άλλο που βρίσκεται μακριά του. Τα λειτουργικά παραδείγματα παρακάτω αποτελούνται από 1 cortical layer. Ένα πιο σύνθετο σύστημα θα μπορούσε να ενώσει εν σειρά πολλά cortical layers.

Οι συνάψεις των HTM νευρώνων είναι επίσης πιστότερες στη βιολογία. Δε χρησιμοποιούν κάποιο βάρος που πολλαπλασιάζει την είσοδο, αλλά είναι δισταθείς, δηλαδή υπάρχουν ή όχι. Ένα μέγεθος όμως που χαρακτηρίζει κάθε σύναψη είναι η *μονιμότητά* της, που διαμορφώνεται δυναμικά από τη μαθησιακή διαδικασία (στην οποία δε θα αναφερθούμε παραπάνω εδώ) με αλγόριθμο τύπου Spike Timing Dependent Plasticity (STDP), που μπορεί ουσιαστικά να εμφανίσει και να εξαφανίσει υποθαλάσσους συνάψεις για να προσαρμόσει το σύστημα στα δεδομένα. Η μονιμότητα μιας σύναψης συγκρίνεται με ένα κατώφλι και, αν είναι πιο πάνω είναι ενεργή, αν είναι πιο κάτω είναι ανενεργή. Μια σύναψη με μονιμότητα λίγο πάνω από το κατώφλι μπορεί δηλαδή να απενεργοποιηθεί εύκολα.

1.3.3 Encoders

Ο εγκέφαλος έχει ως βασική διεπαφή με τον έξω κόσμο τον αμφιβληστροειδή χιτώνα του ματιού και τον κοχλία του αυτιού. Και τα 2 αυτά συστήματα μπορούμε να θεωρήσουμε ότι μετατρέπουν τα φυσικά ερεθίσματα του κόσμου σε μια πρώτη μορφή SDR, για να ερεθίσουν με τη σειρά τους το επόμενο επίπεδο νευρώνων. Το ρόλο αυτό στα συστήματα HTM καλύπτουν οι encoders.

Οι encoders μπορεί να είναι από πολύ απλά έως εξαιρετικά πολύπλοκα συστήματα (βλέπε αμφιβληστροειδή). Στη διαφάνεια 13 δίνεται το παράδειγμα ενός απλού encoder που κωδικοποιεί ακέραιους αριθμούς συγκεκριμένου εύρους. Μπορεί όμως να σχεδιαστεί και encoder που να κωδικοποιεί λέξεις ή και κείμενα, όπως κάνει η εταιρία cortical.io [3] στη διαφάνεια 14.

¹Δες το βίντεο “Topology (Episode 10)” της σειράς “HTM School” [13]

1.4 Βασικές λειτουργίες HTM

Παρουσιάζονται ένα σύστημα που κανονικοποιεί τα SDR εισόδου, ο *Spatial Pooler*, και ένα σύστημα που μαθαίνει να αναγνωρίζει ακολουθίες απλών ερεθισμάτων, δηλαδή μια *μνήμη ακολουθιών*. Για το πρώτο προτείνονται τα σχετικά βίντεο από τη σειρά “HTM School” στο youtube, ενώ για το δεύτερο η αναφορά [2]. Λίγα λόγια για τη μνήμη ακολουθιών:

1.4.1 Sequence Memory

Στη *διαφάνεια 17* [2] βλέπουμε ένα cortical layer που αποτελείται από από minicolumns. Στη μέση (C) βλέπουμε πώς αντιδρά το σύστημα καθώς δέχεται για πρώτη φορά τις αλληλουχίες “ABCD” και “XBCY”. Παρατηρούμε ότι κάθε σύμβολο εισόδου ενεργοποιεί συγκεκριμένα minicolumns (αυτά που διεγείρονται περισσότερο), και ότι σε κάθε τέτοιο minicolumn ενεργοποιούνται *όλοι* οι νευρώνες.

Ας επικεντρωθούμε στο τι συμβαίνει όταν λαμβάνεται η είσοδος “Α”. Ενεργοποιούνται τα minicolumns που φαίνονται, αλλά κάποιοι από τους ενεργοποιημένους νευρώνες έχουν context συνάψεις (ενεργές ή μη) με κάποιους νευρώνες που ενεργοποιεί η είσοδος “Β”. Αυτό φαίνεται στη διαφάνεια 18 Α. Αφού το σύστημα δει την αλληλουχία A -> B μερικές φορές, συνάψεις αυτού του τύπου ενεργοποιούνται μέσω της διαδικασίας εκμάθησης, ακόμα κι αν στην αρχή ήταν ανενεργές. Έτσι, όταν ξαναέρθει η είσοδος “Α”, συγκεκριμένοι νευρώνες των στηλών που αντιστοιχούν στο “Β” τίθενται σε predictive state. Όταν λοιπόν όντως έρχεται η είσοδος “Β” αμέσως μετά αυτοί οι νευρώνες ενεργοποιούνται πρώτοι και μπλοκάρουν μέσω inhibitory συνδέσεων τους υπόλοιπους του ίδιου minicolumn. Έτσι καταλήγουμε στην ακολουθία της κάτω σειράς (D) της διαφάνειας 17. Τα παραπάνω σημαίνουν ότι το σύστημα *προβλέπει την επόμενη είσοδό του*, που είναι ο βασικός στόχος του προβλήματος εκμάθησης ακολουθιών.

Ο μηχανισμός αυτός οδηγεί σε δύο παρατηρήσεις: η εσωτερική αναπαράσταση κάθε συμβόλου είναι *διαφορετική ανάλογα με τα συμφραζόμενα* και το σύστημα μπορεί να πραγματοποιεί *πολλαπλές προβλέψεις ταυτόχρονα*.

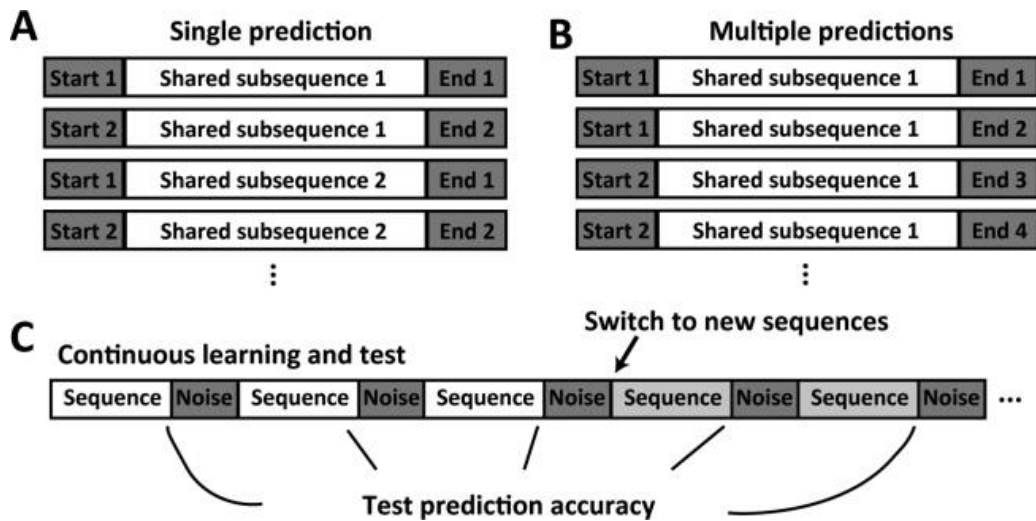
Το πρώτο φαίνεται στην κάτω σειρά (D) από τους διαφορετικούς νευρώνες που ενεργοποιούνται για να συμβολίσουν την ίδια είσοδο “Β”, ανάλογα αν το προηγούμενο σύμβολο ήταν “Α” ή “Χ”, οπότε προκύπτει η εσωτερική αναπαράσταση Β’ ή Β” αντίστοιχα. Το δεύτερο σημείο φαίνεται στη διαφάνεια 18 Β, όπου δίνεται η είσοδος “Β”, χωρίς να έχει προηγηθεί “Α” ή “Χ”. Το σύστημα τότε προβλέπει την ένωση των 2 καταστάσεων C’ και C”, οι οποίες ενεργοποιούνται παράλληλα όταν όντως έρχεται η είσοδος C. Εκείνη τη στιγμή λοιπόν το σύστημα σωστά προβλέπει ότι η συνέχεια της ακολουθίας θα είναι ή “D” ή “Y”.

1.5 Αποτελέσματα - Προσομοιώσεις

Το HTM δοκιμάστηκε τόσο σε τεχνητά, όσο και σε πραγματικά δεδομένα για να αξιολογηθεί η δυνατότητα του να παρέχει ακριβείς προβλέψεις. Παρακάτω θα αναλυθούν δύο παραδείγματα και θα δοθούν τα συγκριτικά διαγράμματα των επιδόσεων με άλλα νευρωνικά δίκτυα.

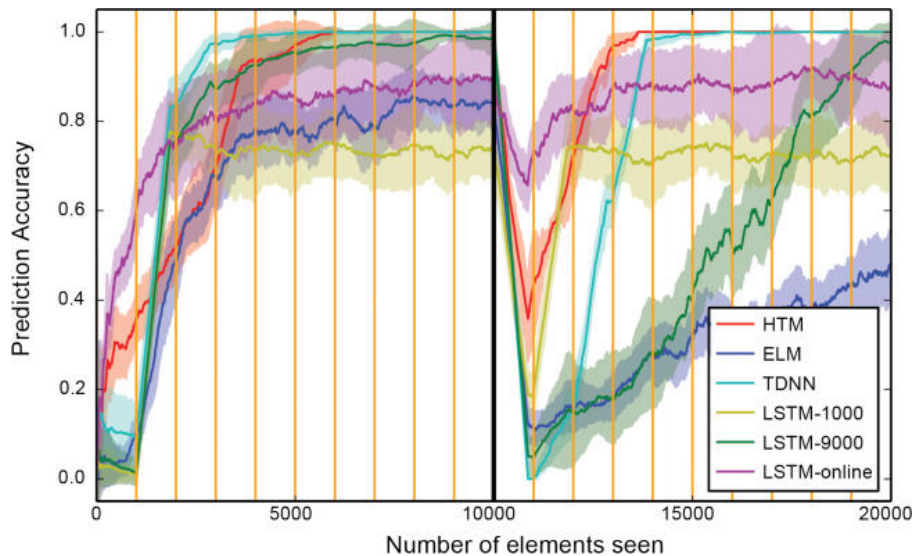
1.5.1 Πρόβλεψη συμβόλου ακολουθίας

Οι αλγόριθμοι του Hierarchical Temporal Memory (HTM) δοκιμάστηκαν σε διαφορετικά προβλήματα [2, 7], ώστε να αξιολογηθεί η επίδοση του συστήματος σε σχέση με άλλες υλοποιήσεις. Στο πρώτο παράδειγμα έχουν δημιουργηθεί δύο ακολουθίες από σύμβολα, σύμφωνα με την εικόνα (1). Στόχος είναι η πρόβλεψη του τελευταίου συμβόλου.



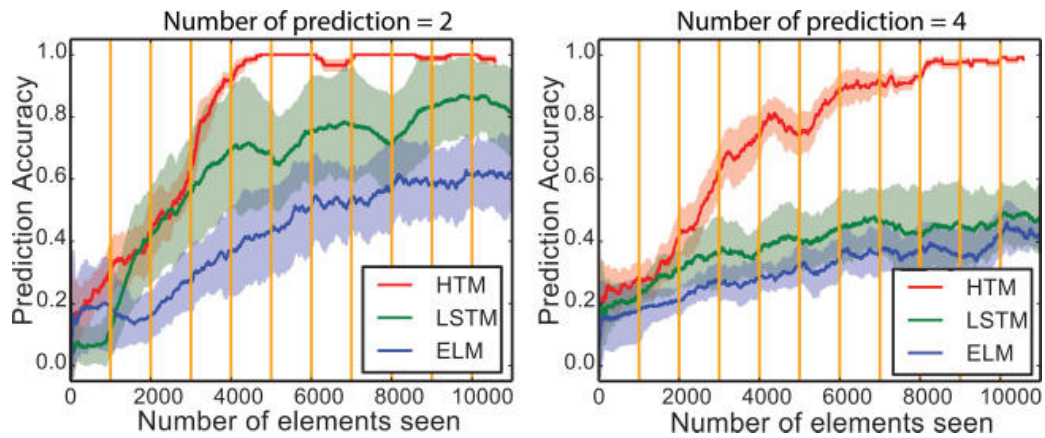
Σχήμα 1: Ακολουθίες συμβόλων για την προσομοίωση

Οι ακολουθίες φτιάχτηκαν έτσι ώστε να χρειάζεται ένα βάθος μνήμης τουλάχιστον 2 παρελθοντικών συμβόλων για την επιτυχή πρόβλεψη. Το πρώτο set ακολουθιών έχει δημιουργηθεί για να ελεγχθεί η δυνατότητα απλής πρόβλεψης, καθώς κάθε ακολουθία προσδιορίζεται πλήρως από τα πεδία “Start” και “Shared Subsequence”. Αντιθέτως, στο δεύτερο set δύο ακολουθίες μπορεί να έχουν ίδια τα παραπάνω πεδία. Σε αυτή την περίπτωση το νευρωνικό δίκτυο θα πρέπει να επιτύχει την ορθή πρόβλεψη όλων των πιθανών καταλήξεων, δηλαδή την ταυτόχρονη πολλαπλή πρόβλεψη. Τέλος, ανάμεσα στις ακολουθίες προστέθηκε ένα σύμβολο θορύβου, ώστε να εξεταστεί η ανθεκτικότητα του συστήματος σε αυτόν.



Σχήμα 2: Απόδοση διαφορετικών υλοποιήσεων για απλή πρόβλεψη

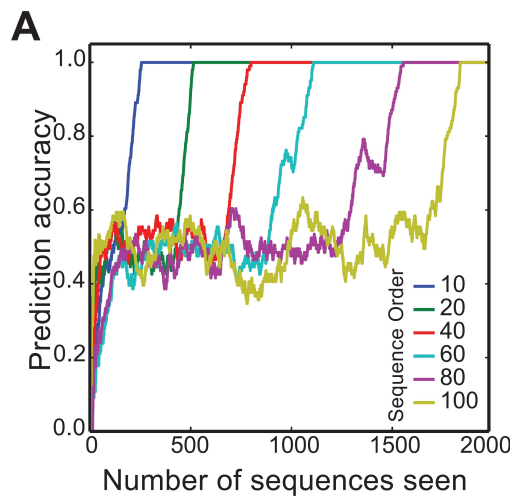
Στο σχήμα (2) δίνεται η συγκριτική απόδοση των διαφορετικών υλοποιήσεων νευρωνικών δικτύων για το πρώτο set ακολουθιών. Τα δίκτυα που υλοποιούνται είναι το HTM, το ELM, το TDNN και το LSTM. Ο προσδιορισμός στο LSTM αφορά το πλήθος των δειγμάτων που χρησιμοποιούνται για retraining σε κάθε κατακόρυφη κίτρινη γραμμή του διαγράμματος. Στο στοιχείο 10000 που σημειώνεται με κάθετη κατακόρυφη γραμμή, αλλάζουν αμοιβαία τα δύο τελευταία σύμβολα κάθε ακολουθίας και μελετάται η προσαρμογή του δικτύου στο νέο input space. Βλέπουμε ότι το HTM πετυχαίνει αρχικά σε έναν ικανοποιητικό αριθμό δειγμάτων ακρίβεια 100%. Το πιο σημαντικό είναι, βέβαια, ότι μέσω online training αναπροσαρμόζεται ταχύτερα από κάθε άλλη υλοποίηση στο νέο χώρο ακολουθιών εισόδου.



Σχήμα 3: Απόδοση δικτύων για πολλαπλές προβλέψεις

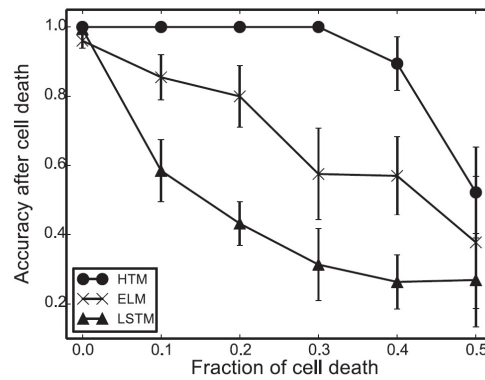
Οι πολλαπλές προβλέψεις εξετάζονται από το δεύτερο σετ και τα συγκριτικά αποτελέσματα για διπλή και τετραπλή πρόβλεψη δίνονται στο σχήμα (3). Όπως φαίνεται, το HTM είναι το μοναδικό σύστημα που πετυχαίνει μετά από κάποιο διάστημα ακρίβεια σχεδόν 100%. Αυτό απορρέει από την αναπαράσταση μέσω SDR και τη δυνατότητα του Temporal Pooler για πολλαπλές προβλέψεις.

Στη συνέχεια, μελετήθηκε η απόδοση του HTM για ακολουθίες διαφορετικού μήκους. Ένα από τα σημαντικότερα στοιχεία του Temporal Pooler είναι η δυνατότητα αναπροσαρμογής online του «βάθους» μνήμης που χρησιμοποιείται για την πρόβλεψη. Έτσι, όπως παρατηρούμε και στο διάγραμμα (4), πετυχαίνει πρόβλεψη ακόμα και τάξης 100, που σημαίνει ότι απαιτείται μνήμη 99 συμβόλων για να προβλεφθεί επιτυχώς το τελευταίο. Το πλήθος των ακολουθιών που απαιτείται για να εκπαιδευτεί το σύστημα και να πετύχει ακρίβεια κοντά στο 100% αυξάνει γραμμικά με την τάξη της ακολουθίας.



Σχήμα 4: Απόδοση HTM συναρτήσει της τάξης της ακολουθίας

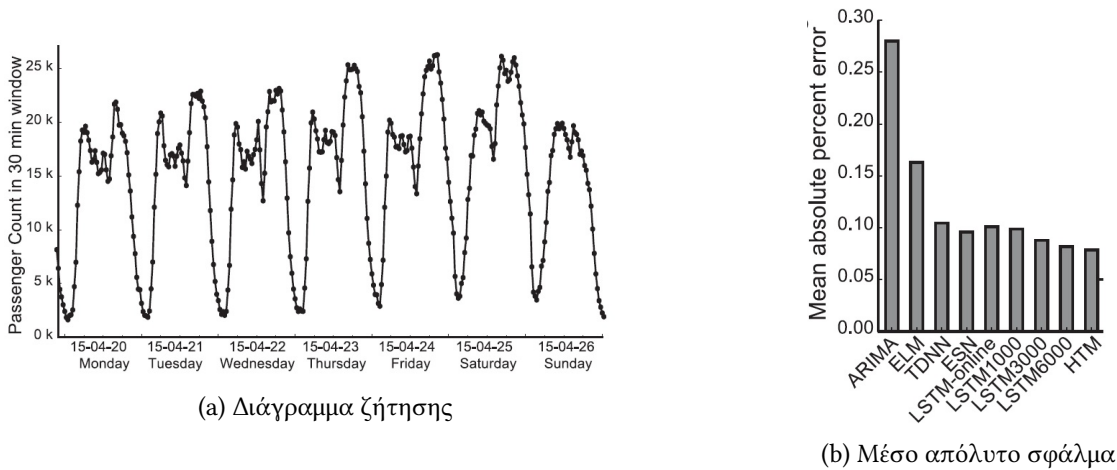
Τέλος, τα διαφορετικά συστήματα εξετάστηκαν και ως προς την ανθεκτικότητα σε καταστροφή του δικτύου μέσω νεκρών νευρώνων. Το HTM έμεινε ανεπηρέαστο ακόμα για 30% νεκρούς νευρώνες, συνθήκες κατά τις οποίες το ELM και το LSTM είχαν υποβαθμιστεί αρκετά.



Σχήμα 5: Ανθεκτικότητα υλοποιήσεων σε νεκρούς νευρώνες

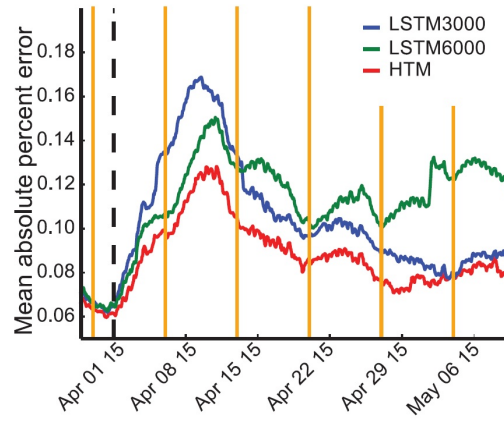
1.5.2 Πρόβλεψη της ζήτησης taxi στη Νέα Υόρκη

Το Hierarchical Temporal Memory δοκιμάστηκε και σε πραγματικά streaming δεδομένα. Πιο συγκεκριμένα, ζητήθηκε η πρόβλεψη της ζήτησης στα taxi της Νέας Υόρκης 2.5 ώρες πριν την πραγματική μέτρηση. Η ζήτηση ποσοτικοποιήθηκε ως η συνολική εξυπηρέτηση πελατών σε ένα χρονικό παράθυρο 30 λεπτών. Εξετάστηκαν διάφοροι τύποι νευρωνικών δικτύων, καθώς και το στατιστικό μοντέλο ARIMA. Όπως παρατηρούμε από την εικόνα (6b), το HTM πέτυχε το μικρότερο απόλυτο σφάλμα πρόβλεψης (της τάξης του 0.08%).



Σχήμα 6: Πρόβλεψη ζήτησης των taxi

Στη συνέχεια, τα δεδομένα εισόδου τροποποιήθηκαν τεχνητά ανεβάζοντας ή μειώνοντας τη ζήτηση κατά 20% σε διαφορετικές χρονικές περιόδους εντός κάθε μέρας. Στόχος ήταν να μελετηθεί η προσαρμογή του δικτύου στις νέες (τεχνητές) συνήθειες των επιβατών taxi της Νέας Υόρκης. Η αλλαγή πραγματοποιήθηκε την 1η Απριλίου και εντός 2 εβδομάδων το σύστημα HTM αναπροσάρμοσε το μοντέλο του, ώστε να πετυχαίνει εκ νέου σωστές προβλέψεις. Σε αντίθεση, το LSTM είτε απέτυχε να επανέρθει, είτε χρειάστηκε πολύ περισσότερο διάστημα για retraining των παραμέτρων του.



Σχήμα 7: Προσαρμογή δικτύων σε μεταβολή της ζήτησης

2 Neuromorphic processors and hardware HTM

2.1 SpiNNaker

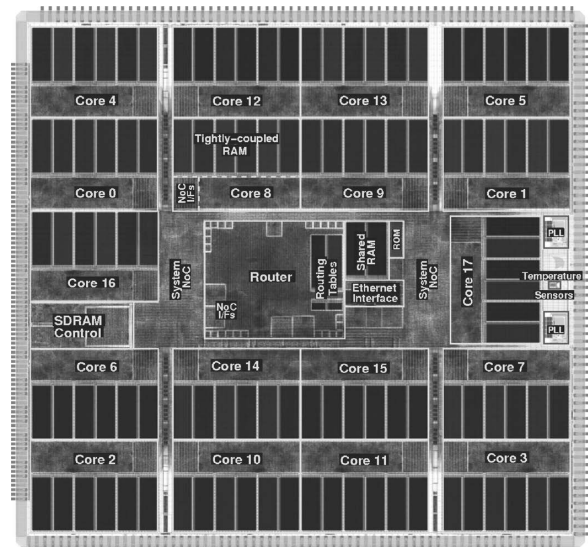
Το SpiNNaker (Spiking Neural Network Architecture) είναι ένα ψηφιακό υπολογιστικό σύστημα υψηλής παραλληλοποίησης κατάλληλο μοντελοποίησης νευρωνικών δικτύων. Αναπτύχθηκε από το Πανεπιστήμιο του Manchester με σκοπό να προσομοιώσει μέρος του ανθρώπινου εγκεφάλου (συγκεκριμένα το 1% των νευρώνων)

2.1.1 Αρχιτεκτονική Συστήματος

Βασικό δομικό στοιχείο του SpiNNaker είναι το SpiNNaker Multiprocessor Chip (CMP). Πρόκειται για chip που αποτελείται από 18 σύγχρονους ARM968 processors που επικοινωνούν μεταξύ τους ασύγχρονα. Είναι κατασκευασμένο σε τεχνολογία CMOS 130nm, λειτουργεί στα 180Hz και καταναλώνει μόλις 1W.

Process	
Process technology	UMC 130 nm 1P8M CMOS
Die area	101.64 mm ²
Power supply	1.2 V (Core), 1.8 V (I/O)
Cores	
Processor cores	18 ARM968s (1 monitor, 16 application & 1 spare)
Processor frequency	180 MHz
Processor node area	3.75 mm ²
Memory	
Local memory per core	32 KB (Instruction), 64 KB (Data)
On-die shared RAM	32 KB
Off-die shared RAM	128 MB DDR2 SDRAM
Communications	
On-chip interconnect	Asynchronous NoCs
Off-chip link b/w	250 Mb/s
On-chip comms Link b/w	5.0 Gb/s
Off-die SDRAM b/w (DMA)	7.2 Gb/s
On-die shared RAM b/w (DMA)	3.2 Gb/s
On-die shared RAM b/w (bridge)	200 Mb/s
Router input b/w	5.3 Gb/s
57K-node sys. bisection b/w	4.6 Gpackets/s
Power Consumption	
Peak (chip)	1 W
Idle (chip)	360 mW
Idle (core)	20 mW
Off-chip link (full speed)	6.3 mW (25pJ/bit)
SDRAM	170 mW
Implementation	
Transistor count	~100 million
Design effort	~40 person-years

(a) CMP specifications

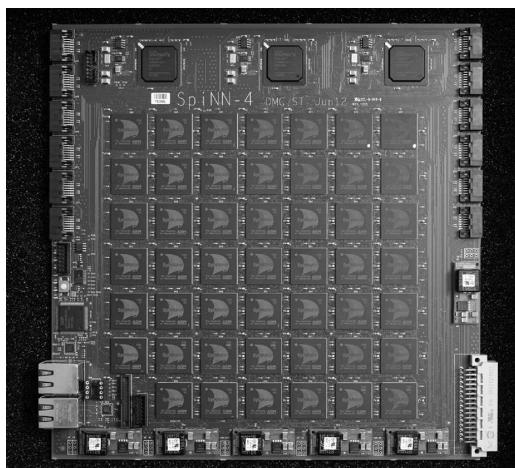


(b) Πλακίδιο CMP

Σχήμα 8: SpiNNaker Multiprocessor Chip (CMP)

Το συνολικό σύστημα αποτελείται από κόμβους που περιέχουν ένα CMP και ένα στοιχείο SDRAM μέσα σε ένα κοινό package. Στην τελευταία ολοκληρωμένη υλοποίηση, υπάρχουν 57600 κόμβοι που περιέχουν συνολικά 10^6 ARM968. Κάθε ARM μπορεί να μοντελοποιήσει περίπου 1000 νευρώνες. Αυτό το πλήθος, βέβαια, μπορεί να ποικίλλει ανάλογα με την πολυπλοκότητα του μοντέλου που χρησιμοποιείται για το νευρώνα. Συνολικά, έχουμε δηλαδή 10^9 νευρώνες που αποτελούν το 1% του ανθρώπινου εγκεφάλου. Η υπολογιστική ισχύς είναι της τάξης των 228 TIPS. Ενδεικτικά, ένας τελευταίας γενιάς i7 της Intel πετυχαίνει περίπου 300GIPS.

Άξια σχολιασμού είναι φυσικά και η κατανάλωση ισχύος ενός τέτοιου μηχανήματος. Παρά τις προσπάθειες για την ελαχιστοποίηση της, το SpiNNaker καταναλώνει περίπου 90KW. Συγκρίνοντας με τα 25W του ανθρώπινου εγκεφάλου, φαίνεται ακόμη περισσότερο το μεγαλείο αυτής της ((μηχανής)) που βιολογικά έχει εξελιχθεί κατά τη διάρκεια ύπαρξης του ανθρώπινου είδους.



Σχήμα 9: Συστοιχία από 48 CMPs

Για την επικοινωνία το Spinnaker χρησιμοποιεί το Address Event Representation (AER). Σύμφωνα με αυτό το πρωτόκολλο, το spike εμπεριέχει μόνο την ταυτότητα του πομπού, ενώ η πληροφορία του χρόνου αποκτάται μέσω της στιγμής αυτό φτάνει στον δέκτη. Όλα τα spikes περνούν μέσα από ένα Network on Chip (NoC) που υπάρχει στο εσωτερικό ενός CMP. Αυτό έχει τη δυνατότητα να αντιγράψει το spike και να το δρομολογήσει στους αποδέκτες.

Σε ότι αφορά τη μνήμη υπάρχει ένα fast access memory για την κατάσταση του νευρώνα εντός του Arm processor. Η κατάσταση του νευρώνα απαιτείται πάντα και για αυτό η προσπέλαση της θα πρέπει να είναι όσο το δυνατόν ταχύτερη. Επιπλέον, υπάρχει και μια SDRAM που περιέχει την κατάσταση των συνάψεων. Οι τελευταίες είναι απαραίτητες μόνον όταν υπάρχει spike στην είσοδο, οπότε μπορούν να υλοποιηθούν και με σχετικά πιο αργή μνήμη.

Όπως αναφέρθηκε, πολύ σημαντικό χαρακτηριστικό τέτοιων υπολογιστικών συστημάτων είναι η κατανάλωση. Για την ελαχιστοποίηση της επιλέχθηκε η ασύγχρονη επικοινωνία που δαπανά λιγότερη ενέργεια από την αντίστοιχης ποιότητας σύγχρονη περίπτωση. Εφαρμόστηκαν, επίσης, τεχνικές sleep mode για τα ανενεργά τμήματα του κυκλώματος, ενώ και η σχεδιαστική επιλογή των ARM968 processors και της SDRAM αποτελεί ένα trade-off μεταξύ απόδοσης και κατανάλωσης. Τέλος, σε ό,τι αφορά την αξιοπιστία εφαρμόστηκαν τεχνικές redundancy, δηλαδή υπάρχουν κάποια εφεδρικά στοιχεία (πχ. buses, NoCs, CMPs) για να μην υπάρξει αποτυχία του κυκλώματος σε περίπτωση βλάβης ή θορύβου.

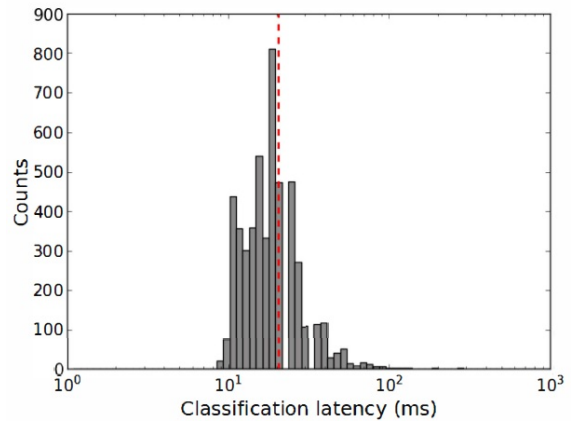
2.1.2 Επίδοση SpiNNaker

Στο SpiNNaker έχουν δοκιμαστεί αρκετά νευρωνικά δίκτυα. Μεταξύ αυτών, ένα Deep Neural Network χρησιμοποιήθηκε για classification χειρόγραφων ψηφίων της βάσης MNIST. Η ακρίβεια του συστήματος συγκρίθηκε τόσο με μια καθαρά μεθόδο λογισμικού (MATLAB), με το Brian που αποτελεί προσομοιωτή του SpiNNaker και το Minitaur που είναι μια άλλη υλοποίηση Hardware. Από την εικόνα (10a) βλέπουμε ότι η υλοποίηση στο Spinnaker υπολείπεται πολύ λίγο σε accuracy από αυτές του software. Ταυτόχρονα,

καταναλώνει μόλις 0.3W ισχύος και έχει σχετικά χαμηλό μέσο χρόνο για το classification (της τάξης των 20ms).

Simulator	CA (%)	Description
Matlab	96.06	Rate-based (Siegert)
Brian	95.07	Clock-driven
SpiNNaker	95.01	Hybrid
Minitaur	92.00	Event-driven

(a) Classification Accuracy



(b) Μέση καθυστέρηση

Σχήμα 10: Επιδόσεις DBN στο SpiNNaker

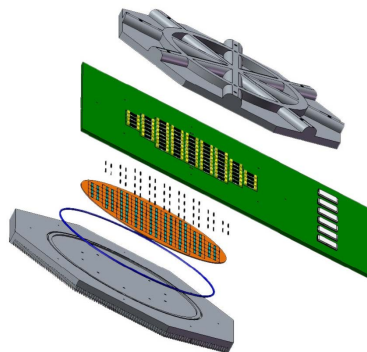
2.2 BrainScaleS Project (BSS)

Στα πλαίσια του BrainScaleS Project δημιουργήθηκε το Heidelberg Neuromorphic Computing Platform. Πρόκειται για ένα mixed-signal υπολογιστικό σύστημα που δίνει τη δυνατότητα προσομοίωσης νευρομορφικών μοντέλων. Αυτό σημαίνει ότι τα chip είναι αναλογικά, ενώ η επικοινωνία τους γίνεται με ψηφιακό τρόπο.

2.2.1 Αρχιτεκτονική Συστήματος

Το BrainScaleS αποτελείται από ένα cluster για την είσοδο και την έξοδο των δεδομένων και ένα νευρομορφικό σύστημα. Πυρήνας αυτού του συστήματος είναι το HICANN (High Input Count Analog Neural Network) Chip. Κάθε HICANN μπορεί να προσομοιώσει μέχρι και 512 νευρώνες τύπου Adaptive Exponential που δέχονται είσοδο από 2 ομάδες των 256 συνάψεων.

Το σύστημα χρησιμοποιεί wafer-scale integration, δηλαδή τα διαφορετικά chip πάνω στο wafer δε κόβονται σε dies, αλλά συνδέονται μεταξύ τους πάνω στο wafer. Στην εικόνα (11) παρατηρούμε το κάτω mounting, ένα προστατευτικό ceil ring, το wafer πυριτίου που περιέχει τα αναλογικά κυκλώματα, τη motherboard πάνω από το wafer και το πάνω mounting. Η motherboard παρέχει τη διεπαφή μεταξύ του wafer και εξωτερικών FPGAs που χρησιμοποιούνται για την επικοινωνία μεταξύ νευρώνων σε διαφορετικά wafer.

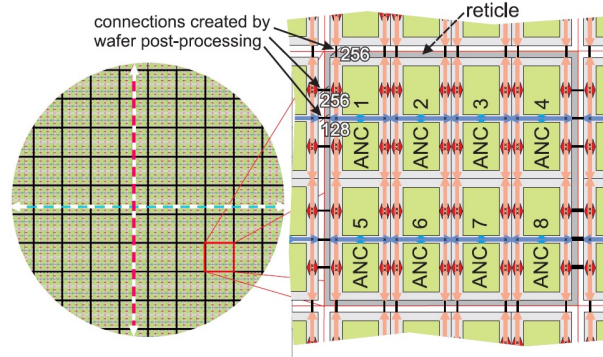


Σχήμα 11: Wafer συστήματος BrainScales

Κάθε wafer περιέχει 384 HICANN chips και μπορεί να μοντελοποιήσει $200 \cdot 10^3$ νευρώνες AdEx με 45 εκατομμύρια συνάψεις. Το σημερινό σύστημα αποτελείται από 20 πλήρως συνδεδεμένα wafer.

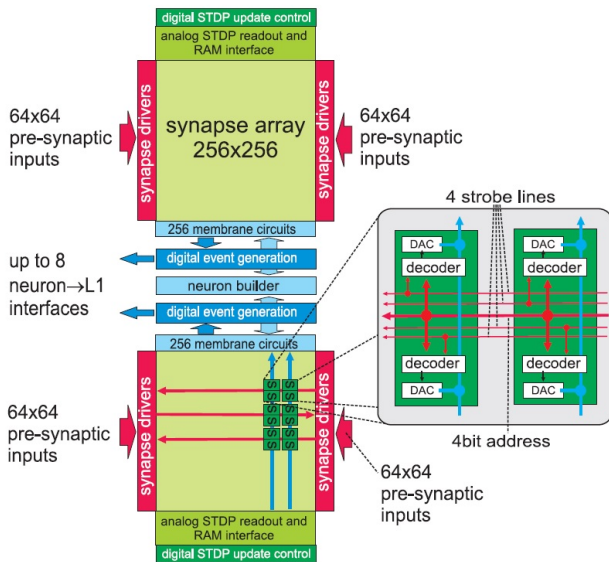
Το wafer χωρίζεται σε reticles, καθένα από τα οποία περιέχει 8 HICANN chips (εικόνα (12)). Τα ANC που σημειώνονται είναι ένας διαφορετικός όρος για το HICANN. Για να επικοινωνούν τα διαφορετικά

HICANNs απαιτείται ένα πυκνό δικτύωμα από οριζόντια και κάθετα wires. Αυτό για ιστορικούς λόγους ονομάζεται Layer 1 routing. Λόγω κατασκευαστικών περιορισμών, τα wires εκτείνονται μόνο εντός του reticle. Η επικοινωνία μεταξύ διαφορετικών reticles εξασφαλίζεται με post-processing στο wafer δημιουργώντας συνδέσεις σε επίπεδο πάνω από το αρχικό.

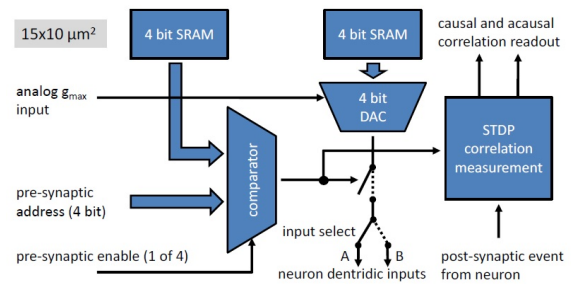


Σχήμα 12: Δομή και διασυνδέσεις ενός reticle

Τα κύρια components του HICANN δίνονται στο σχήμα (13a). Ο πυρήνας του είναι χωρισμένος σε πάνω και κάτω μέρος, με 256 κυκλώματα «μεμβράνης» σε κάθε πλευρά. Αυτά χρησιμοποιούνται για τα μοντέλα των νευρώνων που θέλουμε να υλοποιήσουμε και ανάλογα με την πολυπλοκότητα μπορεί να υπάρχουν περισσότερα membranes/neuron. Αυτή η ομαδοποίηση των κυκλωμάτων μεμβράνης γίνεται μέσω του Neuron Builder. Η είσοδος των spikes πραγματοποιείται μέσω των synapse drivers που χωρίζονται σε 4 ομάδες στο πάνω και κάτω μέρος. Κάθε spike μεταφέρει μια πληροφορία 6 bits, εκ των οποίων τα δύο πρώτα χρησιμοποιούνται για να επιλεγεί κάποιο τα 4 strobe lines (που φαίνονται στη μεγέθυνση). Κάθε strobe έχει μια χρονική διάρκεια παλμού τ_{STDF} που μπορεί να καθορίζεται και από τους learning αλγορίθμους. Τα υπόλοιπα 4 bit μεταφέρονται στο εσωτερικό της διάταξης και αξιοποιούνται στο κομμάτι των συνάψεων.



(a) Δομή HICANN



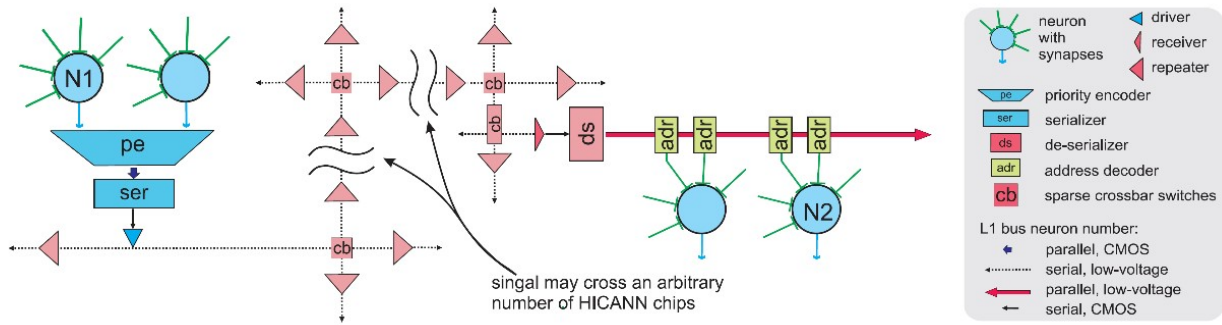
(b) Δομή Σύναψης

Σχήμα 13: Εσωτερικό HICANN chip

Κάθε σύναψη είναι συνδεδεμένη με ένα συγκεκριμένο strobe line. Το βάρος των συνάψεων αποθηκεύεται σε μια 4bit SRAM και μέσω ενός DAC μεταφράζεται σε επίπεδο ρεύματος, όπως φαίνεται στο σχήμα (13b) και πολλαπλασιάζεται με ένα παράγοντα αγωγιμότητας g_{max} που επίσης μπορεί να μεταβάλλεται. Επιπρόσθετα,

κάθε σύναψη περιέχει και ένα address σε μια άλλη 4bit SRAM η οποία συγκρίνεται με τα 4 από τα 6 εναπομείναντα bits που περιέχει το spike. Εφόσον συμπίπτουν, για όσο διάστημα τ_{STDF} διαρκεί ο παλμός του stobe που μπαίνει στο pre-synaptic enable του συγκριτή θα έχουμε input στο δενδρίτη της μεμβράνης. Συνοψίζοντας, έχουμε ένα spike διάρκειας τ_{STDF} και πλάτους $g_{max} \cdot weight$ που μας δίνει ένα πολύ καλό resolution σε ότι αφορά τα weights. Τέλος, υπάρχουν 2 inputs A και B στο κύκλωμα μεμβράνης που μπορούν να επιλεγούν με έναν multiplexer. Το ένα input μπορεί να χρησιμοποιείται ως excitatory και το άλλο ως inhibitory, χρήσιμο στοιχείο για τους HTM αλγορίθμους.

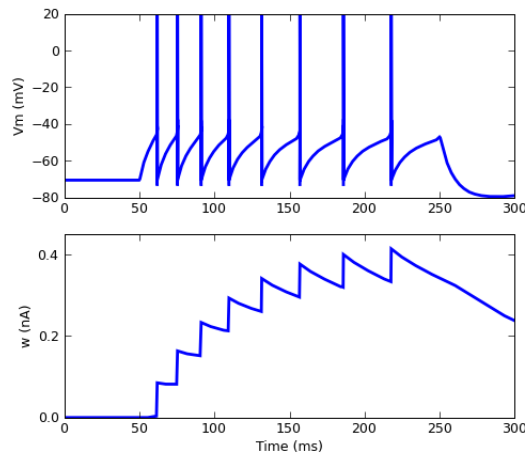
Στη συνέχεια, θα δούμε ένα παράδειγμα επικοινωνίας μεταξύ δύο νευρώνων σε διαφορετικά HICANN chips (εικόνα (14)). Ο priority encoder χρησιμοποιείται για να επιλέξει το neuron με τη μεγαλύτερη προτεραιότητα, ενώ από τον serializer παίρνουμε τη σειριακή ψηφιακή πληροφορία. Στο άκρο κάθε reticle ενσωματώνεται ένας repeater για να αποκαταστήσει το πλάτος και τη διάρκεια του σήματος. Επίσης, σε κάθε διασταύρωση οριζόντιων και κατακόρυφων L1 buses υπάρχουν crossbar switches ώστε το spike να φτάσει τελικά στον προορισμό του περνώντας από διαφορετικά reticles και να αποκωδικοποιηθεί, όπως περιγράφηκε από τον driver του HICANN του δέκτη. Σε περίπτωση που θέλουμε να περάσουμε σε άλλο wafer, θα πρέπει να μεταδοθεί η πληροφορία στη motherboard και με συνδέσεις Ethernet 10 Gbps και FPGA να εισέλθει σε άλλα wafers.



Σχήμα 14: Επικοινωνία HICANNs

2.2.2 Adaptive Exponential (AdEx) Neuron Model

Το σύστημα χρησιμοποιεί το μοντέλο AdEx που υλοποιεί έναν spiking νευρώνα. Αυτό το μοντέλο περιγράφεται από δύο μεταβλητές: το δυναμικό μεμβράνης και το adaptation μέσω ενός συστήματος διαφορικών εξισώσεων. Στην ύπαρξη DC input, το δυναμικό της μεμβράνης αυξάνεται και μόλις γίνει το spike πέφτει στη μηδενική κατάσταση. Το adaptation αναπαριστά το threshold και η αύξηση του κάνει τα spikes όλο και πιο αραιά στο χρόνο (εικόνα (15)).

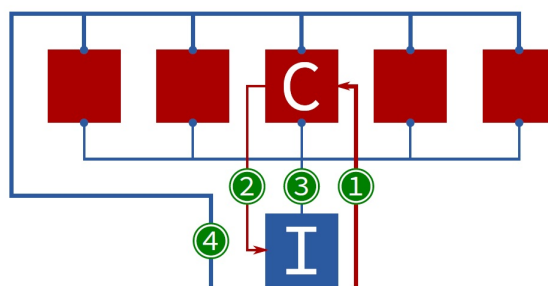


Σχήμα 15: Μοντέλο AdEx Neuron

Στις διαφορικές του μοντέλου υπάρχουν αρκετές παράμετροι που μπορούν να πάρουν κατάλληλες τιμές για να υλοποιήσουν διαφορετικές βιολογικές λειτουργίες. Σημαντικό χαρακτηριστικό αποτελεί ότι το rise time εξαρτάται από την στάθμη της τάσης εισόδου.

2.2.3 Υλοποιήσεις λειτουργιών HTM στο BSS

Τα δύο βασικότερα λειτουργικά μέρη του HTM είναι ο spatial pooler και ο temporal pooler. Στην εικόνα (16) παρατηρούμε την υλοποίηση του spatial pooler, όπου διαφορετικά neurons (με κόκκινο χρώμα) δέχονται μια είσοδο. Όποια έχουν το μεγαλύτερο overlap πυροδοτούνται πρώτα, λόγω μικρότερου rise time. Το inhibition cell (I) δέχεται είσοδο από όλα τα cells και λειτουργεί σαν counter. Μόλις ξεπεραστεί ένα κατώφλι ενεργών cells αποτρέπει τα υπόλοιπα από το να πυροδοτηθούν. Με αυτό τον τρόπο πετυχαίνει να διατηρεί το sparsity της SDR αναπαράστασης. Επιπλέον, έχει τη δυνατότητα σε στιγμές υψηλής συνολικής δραστηριότητας να σταθεροποιεί το σύστημα αυξάνοντας όλους τους χρόνους ανόδου κατά μια σταθερή ποσότητα.



Σχήμα 16: Υλοποίηση Spatial Pooler

Ο temporal pooler προσθέτει το στοιχείο της πρόβλεψης και της ακολουθιακής μνήμης στο HTM. Στα πλαίσια του BSS ένα neuron ενός minicolumn απαιτεί μια τριάδα από cells (D,I,S), όπως στην εικόνα (17). Όλα τα neurons του minicolumn μοιράζονται το ίδιο perceptive field μέσω του στοιχείου P. Στην πραγματικότητα το P αποτελεί μέρος του spatial pooler που περιγράφηκε στην προηγούμενη παράγραφο. Σε περίπτωση που δεν υπάρχει predicted cell και το minicolumn λάβει επαρκή είσοδο, μέσω του P θα ενεργοποιηθούν όλα τα σώματα S. Αντίθετα, αν υπάρχει ένα predicted cell, το I και D cell είναι σε μια depolarized κατάσταση. Μόλις έρθει input το I θα εμποδίσει μέσω inhibitory σήματος την πυροδότηση των υπόλοιπων σωμάτων. Τέλος, για να επιτραπούν πολλαπλές προβλέψεις εντός του minicolumn συνδέεται και το D με το S, ώστε να αναιρέσει στην επίδραση του inhibition σε περίπτωση 2 ή περισσότερων predicted cells.

3 Βιβλιογραφία

Αναφορές

- [1] S. Billaudelle and S. Ahmad. “Porting HTM Models to the Heidelberg Neuromorphic Computing Platform”. In: *ArXiv e-prints* (May 2015). arXiv: 1505.02142 [q-bio.NC].
- [2] Yuwei Cui, Subutar Ahmad, and Jeff Hawkins. “Continuous online sequence learning with an unsupervised neural network model”. In: *Neural Computation* (2016).
- [3] F. De Sousa Webber. “Semantic Folding Theory And its Application in Semantic Fingerprinting”. In: *ArXiv e-prints* (Nov. 2015). arXiv: 1511.08855 [cs.AI].
- [4] Jeff Hawkins and Subutai Ahmad. “Why Neurons Have Thousands of Synapses, a Theory of Sequence Memory in Neocortex”. In: *Frontiers in Neural Circuits* 10 (2016), p. 23. ISSN: 1662-5110. DOI: 10.3389/fncir.2016.00023. URL: <http://journal.frontiersin.org/article/10.3389/fncir.2016.00023>.
- [5] Max Jaderberg et al. “Reinforcement Learning with Unsupervised Auxiliary Tasks”. In: *CoRR* abs/1611.05397 (2016). URL: <http://arxiv.org/abs/1611.05397>.
- [6] Pentti Kanerva. “Associative Neural Memories”. In: ed. by Mohamad H. Hassoun. New York, NY, USA: Oxford University Press, Inc., 1993. Chap. Sparse Distributed Memory and Related Models, pp. 50–76. ISBN: 0-19-507682-6.
- [7] A. Lavin and S. Ahmad. “Evaluating Real-Time Anomaly Detection Algorithms – The Numenta Anomaly Benchmark”. In: *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. Dec. 2015, pp. 38–44. DOI: 10.1109/ICMLA.2015.141.
- [8] Yann LeCun. “Deep learning”. CERN talk. 2016. URL: <https://indico.cern.ch/event/510372/attachments/1245509/1840815/lecun-20160324-cern.pdf>.
- [9] Eustace Painkras et al. “SpiNNaker: A 1-W 18-core system-on-chip for massively-parallel neural network simulation”. In: *IEEE Journal of Solid-State Circuits* 48.8 (2013), pp. 1943–1953.
- [10] Johannes Schemmel, Johannes Fierens, and Karlheinz Meier. “Wafer-scale integration of analog neural networks”. In: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. 2008, pp. 431–438.
- [11] Silver David et al. “Mastering the game of Go with deep neural networks and tree search”. In: *Nature* 529.7587 (Jan. 2016), pp. 484–489. ISSN: 0028-0836. DOI: <http://dx.doi.org/10.1038/nature16961>. URL: <http://www.nature.com/nature/journal/v529/n7587/abs/nature16961.html%5C#supplementary-information>.
- [12] Soniya, S. Paul, and L. Singh. “A review on advances in deep learning”. In: *2015 IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions (WCI)*. Dec. 2015, pp. 1–6. DOI: 10.1109/WCI.2015.7495514.
- [13] Matt Taylor. *HTM School*. Video lectures on YouTube. 2016. URL: <https://www.youtube.com/playlist?list=PL3yXMgtrZmDqhsFQzwUC9V8MeeV0Q7eZ9>.
- [14] C. M. Vineyard and S. J. Verzi. “Overcoming the Static Learning Bottleneck - the need for adaptive neural learning”. In: *2016 IEEE International Conference on Rebooting Computing (ICRC)*. Oct. 2016, pp. 1–3. DOI: 10.1109/ICRC.2016.7738692.
- [15] G. Volanis et al. “Toward Silicon-Based Cognitive Neuromorphic ICs - A Survey”. In: *IEEE Design Test* 33.3 (June 2016), pp. 91–102. ISSN: 2168-2356. DOI: 10.1109/MDAT.2016.2545159.