Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 140 (2018) 134–143

Procedia Computer Science

www.elsevier.com/locate/procedia

Complex Adaptive Systems Conference with Theme: Cyber Physical Systems and Deep Learning, CAS 2018, 5 November – 7 November 2018, Chicago, Illinois, USA

Unsupervised Learning of Polychronous Wavefront Computation Configurations for Pattern Recognition

Fred Highland[a]*

[a]*University of Maryland Baltimore County, Baltimore, MD, USA*

Abstract

Polychronous Wavefront Computation (PWC) provides a potentially simple model for large scale implementation of spiking neural networks and deep learning. Although the definition of predefined pattern recognition configurations has been demonstrated, dynamic organization of configurations from examples remains a difficult problem. This paper explores the hypothesis that a properly arranged field of PWC transponders with neuromorphic behaviors can self-organize into recognition configurations based on training examples. The PWC transponders used are augmented with a position learning algorithm based on spike-timing-dependent plasticity, suppression of non-specific transponders using a stimulation fatigue approach and deactivation of unused transponders using potentiation decay. The paper provides the results of initial research demonstrating that pattern recognition configurations can be learned if the initial density and distribution of transponders is properly selected with respect to the learning behavior parameters. The effectiveness of the learning process can be improved by encoding layering information in the wavefronts to focus the transponder activations. The results define a means for PWC transponders to self-organize into recognition configurations providing a basis for development of more complex configurations and deep learning applications.

Keywords: Unsupervised Learning, Spiking Neural Network, Polychronous Wavefront Computation

* Corresponding author. Tel.: +0-301-471-4685.
E-mail address: fred.highland.mobile@gmail.com

10.1016/j.procs.2018.10.311

1. Introduction

Deep neural networks utilizing thousands of artificial neurons have proven to be effective for complex classification tasks such as speech and image recognition [1][2]. These networks have been implemented on classical von Neuman architectures using multi-core, multi-processor or graphic processing unit approaches to attain the performance and parallelism needed for these systems to be effective. In order to further realize the full potential of artificial neural networks to address complex problems, an architecture utilizing large numbers of simple processing units is required.

Polychronous Wavefront Computation (PWC) [3] was proposed as an abstraction of the spiking neural network paradigm [4]. It is based on the use of transponders (an analog for artificial neurons) that react to temporal and spatial patterns of wavefront activity in a pulse propagating media. This provides a potentially practical model for large scale neuromorphic computing systems because of its simple design that eliminates the need for direct connections between computational units reducing the complexity of implementation.

Izhikevich and Hoppensteadt [3] defined the basic concept of PWC and gave examples of small transponder configurations that can perform computations such as signal analysis and logical operations. The mathematical properties of simple PWC configurations have been explored in detail by Thomas [5] and some conceptual sensor configurations suggested [6]. It has also been suggested that numerical programming methods could be used to configure PWC transponders [7] but details of this approach have not been explored. A basis for unsupervised learning of individual transponder positions has been demonstrated [8] but has not been applied to the generation of larger configurations that could perform pattern recognition. In addition, various transponder design patterns have been defined [9][10] that can be used to perform pattern recognition but the configurations must be precomputed based on the target patterns limiting their utility for the general class of unsupervised recognition problems.

The objective of this work is to define methods to support dynamic self-organization of pattern recognizing PWC transponder configurations from examples (i.e., unsupervised learning). It is desirable that these methods incorporate a homogeneous transponder model to maximize generality and adaptability. The research builds on many of the concepts from prior work to examine the hypothesis that a properly arranged field of PWC transponders with neuromorphic behaviors can self-organize into recognition configurations based on training examples.

The paper will present the preliminary results from research on creating pattern recognition configurations from a field of PWC transponders. It will begin with an overview of the computational model used including a review of the basic PWC concept, extensions to those concepts based on biological analogues and the effect of model parameters on transponder ensemble behavior. The model will then be experimentally applied to a simple pattern recognition problem to determine the characteristics and effectiveness of the approach. The results from those experiments will then be analyzed, the findings will be summarized and directions for future work suggested.

2. Computational Model

The recognition architecture used is based on the multi-layer perceptron paradigm [11] implemented with PWC transponders. This consists of three logical layers of transponders: an input layer that represents features (stimulus), a hidden layer that associate sets of features into higher level concepts and an output layer that combines concepts into recognition signals. The hidden layer may be more than one level deep to represent more complex concepts. The layers communicate through wavefronts emitted by the transponders (explained in more detail below). The multilayer perceptron model has known limitations as a recognition mechanism but serves as a basis for experimentation that can be extended to more complex structures (e.g. deep neural networks).

The starting hypothesis is that a properly arranged field of PWC transponders with the proper behaviors can self-organize into recognition configurations based on training examples. The input layer transponders are arranged in fixed locations to emit wavefronts that represent features of an input pattern (e.g., pixel values in an image). Input layer wavefronts are detected by the hidden layer transponders which modify their positions to coincide with input wavefront intersections in order to emit hidden wavefronts representing sets of input features. The hidden layer wavefronts are then detected by output layer transponders which modify their positions to reliably produce recognition wavefronts.

In order to learn a recognition configuration from examples, the transponders require the following capabilities:

- Discrimination of wavefronts in configurations with a large number of transponders.
- Adjustment of transponder position to optimize detection
- Suppression of transponders that do not participate in the recognition process
- Suppression of output transponders if they do not discriminate unique input patterns
- An initial transponder topology that is conducive to learning the possible input patterns

Details of the basic concepts of PWC and extensions required for unsupervised learning are discussed below.

2.1. Polychronous Wavefront Computation

Polychronous Wavefront Computation (PWC) [3] provides a simplified model of spiking neural networks consisting of a configuration of transponders that may emit wavefronts and sense the wavefronts from other transponders. When a transponder senses two (or more) wavefronts simultaneously, it emits a new wavefront.

Figure 1 shows an example of PWC using a multi-layer pattern recognition design. The topology consists of three layers of transponders: Input, Hidden and Output. The figure shows a time sequence of transponder activation and wavefront propagation. At time t=3, input transponders I1, I2 and I3 have emitted wavefronts (red dashed lines) at different times. At time t=4 the wavefronts from transponders I1 and I2 intersect at hidden transponder H1 causing it to activate. At time t=5 the wavefronts from transponders I2 and I3 intersect at hidden transponder H3 causing it to activate while the wavefront from H1 (blue dotted line) propagates. At time t=9 the wavefronts from transponders H1 and H3 intersect at output transponder O2 resulting in an activation indicating recognition of an input pattern.

The locations of wavefront intersections created by two transponders are defined by a hyperbola based on the radii of the wavefronts that extends outward to infinity from the area between the transponders. In complex configurations, the number of intersection hyperbolae increases combinatorically with the number of transponders, potentially producing a large number of unwanted transponder activations resulting in chaotic behavior. This can be controlled by careful spacing and choice of position but the number of intersection hyperbolae in complex configurations makes this approach difficult to manage. A better approach, which is analyzed by Thomas [5], is to use criteria based on three wavefronts intersections. The intersection of three wavefronts is defined by the intersection hyperbolae of each of the three transponder pairs. This is a single point in space significantly limiting the possibility of unwanted transponder activations and increasing the potential information density of the intersections and the configuration.

2.2. Position Learning

It may be possible to determine the transponders necessary for pattern recognition from an initial field of fixed transponders of sufficient quantity and density simply by eliminating unused transponders. However, the initial number of transponders would be very large and the number of indiscriminate activations high, presenting computational and physical limitations with this approach. Experience has shown [9] that most of these transponders are not needed.

A more effective approach would be to allow a small set of transponders to learn the best positions based on the sensing of potentially activating wavefronts. A method for position learning was defined in previous work [8] which is based modifying the transponder model to incorporate a form of Spike Timing Dependent Plasticity (STDP) [12] and adjusting the new transponder location based on the prediction of wavefront intersections. In biological neurons, STDP works by adjusting the synaptic weights to reinforce stimuli that result in activation. Since the PWC synaptic analogue (wavefronts) is not specific to a stimulating transponder, an alternative method is required. This can be accomplished by adjusting the transponder position in the direction of the predicted intersection of stimulating wavefronts. The position learning approach also generalizes the discrete PWC intersection criteria using a Leaky Integrate-and-Fire Model [13][14] that integrates the strength of multiple wavefronts and applies an exponential decay adjustment based on relative arrival time to create a continuous signal model. This approach allows the detection of "close" intersections providing a basis for iterative location adjustment when the integrated stimulus exceeds a threshold ($\vartheta_{membrane}$) according to:

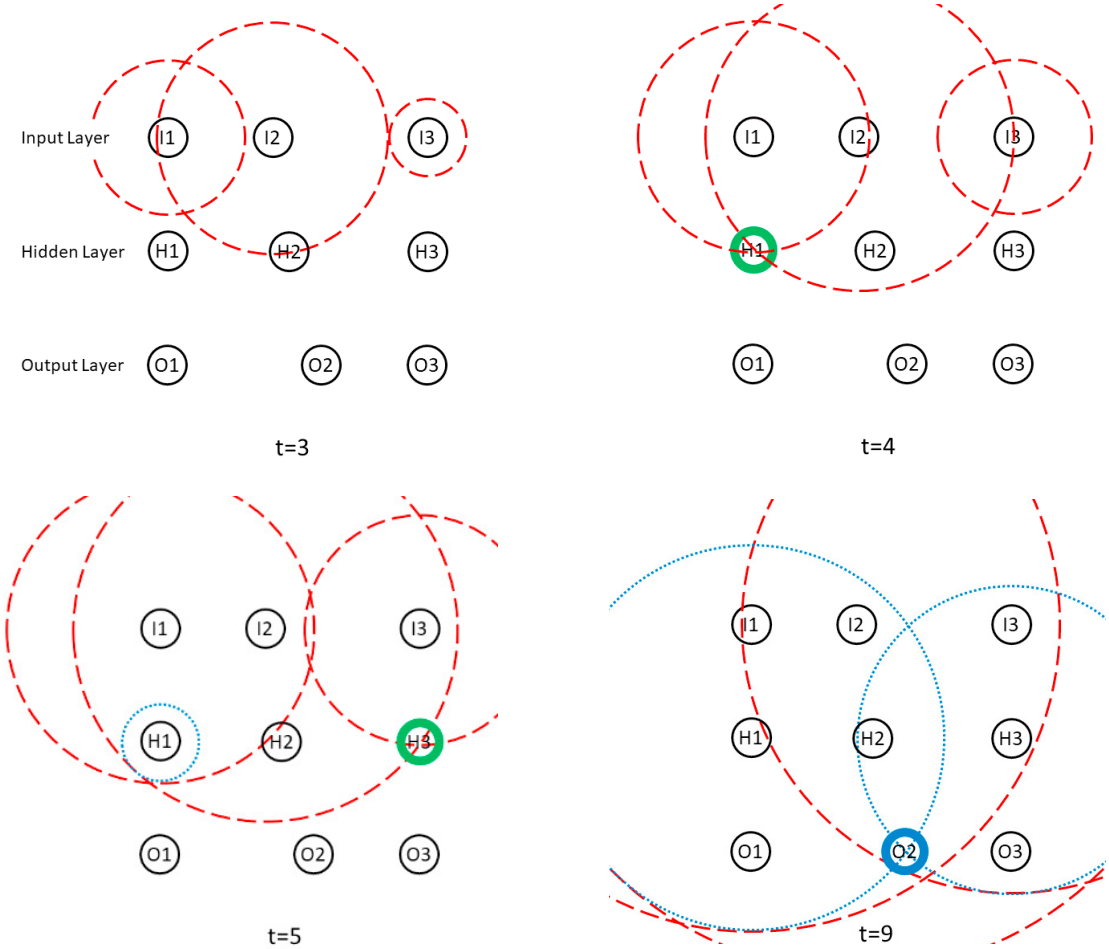


Figure 1 – A Time Sequence of PWC Transponders in a Multi-layer Pattern Recognition Configuration: t=3: I1, I2 and I3 input transponders activate. t=4: wavefronts from I1 and I2 activate hidden transponder H1. t=5: wavefronts from I2 and I3 activate hidden transponder H3. t=9: wavefronts from hidden transponders H1 and H3 activate output transponder O2.

$$A_{+}e^{-(\tau-\tau_0)/\tau_{+}} \tag{1}$$

Where:

A_+ - maximum position modification (potentiation limit)
t - time of transponder activation
t_0 - time of pulse receipt at the transponder
τ_+ - potentiation time constant

The original proposed approach [8] assumed a two wavefront intersection allowing the use of simple line intersection calculations. The prediction method has been extended by determining the centroid of the intersections of contributing pairs of wavefronts and predicting the best new location using gradient decent. This method incrementally improves the position of the transponder and generalizes the method to multiple wavefront intersections.

Although the use of a three wavefront intersection criteria greatly improves the specificity of wavefront detection, the potentially large number of activations in the multi-layer transponder environment can result in superfluous activations that inhibit the recognition process. These can occur from the intersections of hidden and output wavefronts

that do not provide useful information for pattern recognition. In order to eliminate unnecessary interactions, filtering mechanisms are added to the intersection criteria in each transponder. Wavefront sensing is first filtered by wavefront class based on the originating transponder. That is, hidden transponders only sense wavefronts from input transponders and output transponders only sense wavefronts from hidden transponders. This filtering is sufficient for hidden transponders since the input wavefronts represent discrete features. This is not the case for output transponders as multiple hidden wavefronts representing overlapping input features can occur. To address this issue a second level of filtering is used that specifies the minimum number of unique features that must be represented by the wavefront intersections. Using wavefront class and unique feature count filtering, rather than specific transponder identifications, preserves the generic nature of transponders and avoids assignment of a receiving transponder to specific input feature sets or hidden transponders.

The position learning algorithm implicitly defines a detection probability space around the current transponder based on the behavior parameters of the transponder. According to the position learning algorithm [8] transponder position is adjusted only if the sum of the stimulus is greater than or equal to the construction threshold, i.e.,

$$\sum_{i=1}^{n} e^{-\Delta t_i/\tau_{synapse}} \geq \vartheta_{construction} \tag{2}$$

Where

$n_{wavefronts}$ – the number of intersecting wavefronts
Δt_i – the time from wavefront 1 to wavefront i
$\tau_{synapse}$ – the synapse time constant
$\vartheta_{construction}$ – the threshold for learning

Assuming the wavefront propagates at a single distance unit per time unit and the most recent wavefront ($i=1$) provides a stimulus of 1, the radius ($r_{detection}$) of the detection probability space is given by computing the expected value of Δt_i ($\overline{\Delta t}$) found by rearrangement of equation 2 as:

$$r_{detection} = \overline{\Delta t} = -\ln\left(\frac{\vartheta_{construction} - 1}{n_{wavefronts} - 1}\right)\tau_{synapse} \tag{3}$$

The value of $r_{detection}$ provides a basis for determining the minimum spacing between transponders in the initial transponder field. Since the position learning algorithm incrementally relocates transponders, the total range of transponder movement may be larger than $r_{detection}$ depending on a complex interaction of a number of other model parameters including the training sequence and the movement of other transponders. $r_{detection}$ provides a starting point for estimating the density of transponders needed but additional scaling may be needed.

2.3. Potentiation Decay

In the example of Figure 1, hidden node H2 and output nodes O1 and O3 are never activated. For an initial field of transponders that can learn any possible set of input patterns, there will likely be many possible intersections (and transponders that represent them) that will never be needed. It is desirable to dynamically suppress these unnecessary transponders to make the overall configuration operate more efficiently. This can be done by applying the concept of potentiation decay [15]. In biological neural networks, potentiation decay is represented by adjusting the weights of infrequently stimulated synapses. Since PWC transponders do not have synaptic weights an equivalent effect can be achieved by rescaling the activation threshold ϑ over the time interval t by a constant β according to:

$$\vartheta_{n+t} = \vartheta_n/\beta^t, \quad 0 < \beta < 1 \tag{4}$$

This results in an increase in the activation threshold value (decrease in sensitivity) over time, leading to deactivation of the transponder [10].

Transponder desensitization from lack of stimulation must be complimented with an increase sensitivity when activated. This can be achieved by applying a modification of Spike Timing Dependent Plasticity [12] to the threshold level:

$$\vartheta_{n+1} = \vartheta_n - A_{gain} e^{-\Delta t/\tau_{gain}} (\vartheta_n - \vartheta_{min}) \tag{5}$$

Where:

ϑ_n – the current activation threshold
ϑ_{min} – the threshold lower limit
A_{gain} – membrane gain limit
Δt – the time since the last transponder activation
τ_{gain} – the potentiation time constant

The choice of β should be made based on the training time interval as unstimulated transponders should become deactivated in that timeframe. Assuming a training time of $t_{training}$, a default (initial) activation level of $\vartheta_{Default}$ and the number of wavefronts required for activation ϑ_{max}, the target for β can be attained by rearrangement of equation 4 as:

$$\beta = \sqrt[t_{Training}]{\vartheta_{Default}/\vartheta_{Max}} \tag{6}$$

2.4. Synaptic Fatigue

For complex pattern recognition problems, it is possible for multiple wavefront intersections to occur within the probability space defined in 2.2. As a result, some transponder activations may not be specific to a single input pattern providing incorrect results. This problem may be eliminated by applying the concept of synaptic fatigue [16] in which synapses are desensitized by excessive frequent stimulation.

Synaptic fatigue can be simulated in PWC transponders by representing an activation charge resource ($Charge_{Activation}$) which is depleted of by a fixed amount ($Cost_{Activation}$) for each activation and recharged over time [10]. That is:

$$Charge_{Activation} = Charge_{Activation} - Cost_{Activation} \quad where: Charge_{Activation} \geq Cost_{Activation} \tag{7}$$

If the activation frequency is greater than the charge rate, the activation charge resource will eventually be insufficient to support activation. This works in conjunction with Potentiation Decay (Section 2.3) to eventually suppress the transponder through inactivity. The previous work defined the charging method as an exponential process but experiments suggested this charged too quickly. It was found that modification of the charging rate by the square of the current change level produced better results. The resulting activation resource charging occurs according to:

$$Charge_{Activation} = Charge_{Activation} + (1 - Charge_{Activation})\left(1 - e^{-t/t_{Charge}}\right) Charge^2_{Activation} \tag{8}$$

Where

$Charge_{Activation}$ is the current activation resource level
t is the time since last charge
t_{Charge} is the activation charge time constant

The values for $Cost_{Activation}$ and t_{Charge} are closely related to the frequency of repetitions used in training. Setting $Cost_{Activation}$ to $1/n_{example}$ allows a sequence of examples to be learned and suppresses subsequent activations of transponders that are not specific. The setting of t_{Charge} to approximately $t_{example} n_{example}/e$ seemed to produce good results but the success of results is very sensitive to the value of t_{Charge} and the value must be chosen carefully.

Synaptic fatigue is a useful mechanism for suppression of non-discriminating output transponders which are likely to occur. However, hidden transponders react to feature subsets occurring in many input patterns frequently during the recognition process. Therefore, this mechanism should be suppressed or the recharge rate significantly increased (a low value of t_{charge}) for hidden transponders to prevent interference with their normal operation.

2.5. Transponder Topology

The initial topology of transponders is the key input to the process of determining the positions of transponders for recognition. They must be arranged to provide sufficient transponders to accomplish recognition, while minimizing

the number of transponders to prevent non-specific activations, optimize implementation costs and optimize performance.

Analysis done as part of prior work [6][9] suggested a circular arrangement of input transponders representing pattern features using a three wavefront intersection activation criteria is the best approach. The intersection region of three transponders is roughly the interior of a triangle formed by the transponders suggesting a circular pattern for input transponder arrangement. If the transponders are arranged in a circle with offset activation times, the intersections occur generally in the interior of the circle and can maximize unique intersection locations with the right choice of input positions and time offsets

Figure 2 shows the frequency distribution of hidden and output transponder positions with respect to distance from the center using a circular input pattern with a radius of 100 units. The hidden transponder frequency is approximately exponentially distributed with respect to distance from the center. This is expected as the circular input arrangement is likely to produce wavefront intersections toward the center of the circle. The hidden transponder activations result in output transponder activation from the center outward with a heavier tail the more closely follows a gamma distribution. The distribution of intersection density is not directly useful as the transponder positions needed for recognition cannot be easily predicted. However, it does indicate that there is a higher likelihood of useful intersections in a range between the center and the input transponder circle. In addition, the decrease in density with respect to the radius from the center suggests that changing the transponder density and probability region radius may minimize the number of transponders needed.

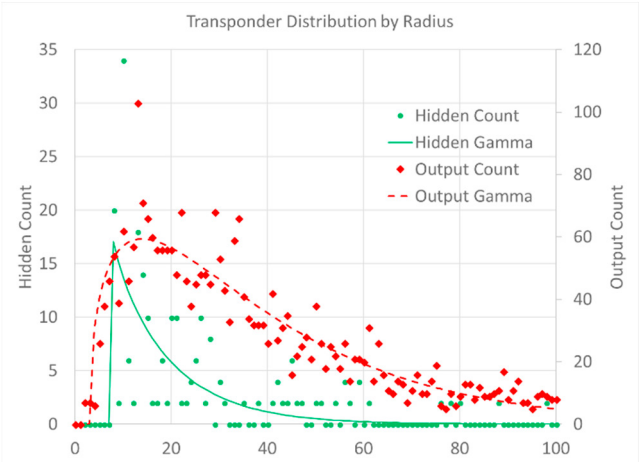


Figure 2 - Distribution of Hidden and Output Transponders for 7-Segment Digit Recognition.

To represent these observations, the initial transponder topology is generated in a circular arrangement of concentric rings, limited by a minimum (r_{min}) and maximum (r_{max}) radius derived from transponder distribution analysis. The spacing between the transponders rings is derived iteratively from the radius of the probability region adjusted by a dilation factor (D_{radial}) and a power (P_{radial}) of the ratio of the current ring radius to the minimum radius. That is:

$$r_{ring\ n+1} = r_{min} + \left({}^{r_{detection}}/_{D_{radial}}\right)\left({}^{r_{ring\ n}}/_{r_{min}}\right)^{P_{radial}} \tag{9}$$

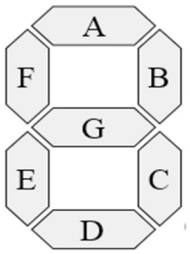
Adjusting the dilation factor and power value allows different radial transponder densities to be generated. Within a ring, the density can be further adjusted by applying an angular density ($D_{angular}$) factor to the radial density. Since the detection of wavefronts and position adjustment is a function of transponder parameters, $\tau_{synapse}$ is also adjusted by the density factors to match transponder sensitivity to the density. Separate parameters are used for hidden and output transponder generation to represent the different characteristics of those transponder roles.

3. Experimental Design

To evaluate the concepts defined above, the transponder behavior, initial transponder topology generation, test scenario management and analysis tools were implemented on a PWC simulator built with NetLogo [17]. The simulator implements PWC transponders in a dynamic, agent-based, discrete-event simulation that allows the wavefront and transponder actions to be visualized, experiments with parameter values to be conducted and performance data to be collected. The simulator supports Leaky Integrate-then-Fire, position learning, potentiation decay and synaptic fatigue semantics to experiment with the effects of these behaviors on the learning process.

A simple but representative pattern recognition problem based on 7-segment display digit recognition was used to evaluate the learning concepts. 7-segment display devices have been used to display decimal digits on electronic devices since about 1910 [18]. They are driven by decoders that convert binary numbers to a decimal representation of seven illuminated segments. The problem used here is categorize a 7-segment pattern (the standard notation is shown in Figure 3) into one of ten outputs labeled 0-9. This problem represents a simple form of a multi-feature classification problem providing a practical basis for detailed analysis and experimentation.

The inputs were encoded as seven features of two mutually exclusive values (segment "on" or "off") represented by 14 transponders. Input transponders positions were evenly distributed around a circle with a 100 unit radius. The time delay between each input feature's transponders was defined as 6 units. The "off" transponders were positioned on the opposite side of the circle from the "on" transponders but activated at the same time since the feature values are mutually exclusive. Placing the "on" and "off" transponders on the opposite side of the circle maximizes the distribution of intersections and minimizes conflicts.



Decimal	Segments
0	ABCDEF
1	BC
2	ABDEG
3	ABCDG
4	BCFG
5	ACDFG
6	ACDEFG
7	ABC
8	ABCDEFG
9	ABCDFG

Figure 3 - Standard Label Nomenclature for a 7-Segment Display.

The initial positions of transponders were generated using the concentric circle method defined above. The simulation was then stepped through a training and verification scenario to learn the best positions for the transponders and analyze the results.

Tests were run with a series of patterns to determine the effectiveness of learning methods. The test cases consisted of repeated series of positive training examples representing the digits 0-9 followed by verification cases consisting of all valid digit patterns. Each digit pattern case spanned 500 time units to allow time for wavefront propagation to complete. Training cases consisted sequences of 10 examples of each valid digit repeated twice.

Data was collected using the visualizations provided by the simulator for verification of overall behavior followed by detailed analysis of the effects of configuration parameters using NetLogo's Behavior Space tool. Further analysis supported with Excel spreadsheets. Behavior Space provides a facility to specify ranges of model parameters, run the models in parallel as background tasks and record the results for post simulation analysis. The ability to specify ranges of parameter values and run multiple instances of the simulation in parallel make it an effective tool for understanding and optimizing the behavior of complex models with a large number of possible parameter values.

The results were evaluated against a set of success and optimization criteria to determine effectiveness. The primary success criterion was reliable recognition of the test patterns by associating input patterns with one or more output transponders. The corollary to this criterion was the suppression of output transponders that recognize multiple patterns (non-specific recognition). Suppression of output transponders that do not participate in recognition is also desirable but not strictly required. Having a minimum number of transponders is preferred from both a performance and implementation standpoint. A single output transponder per pattern would be ideal but multiple transponders per pattern are acceptable as long as they correctly discriminate input patterns. Attaining a minimum number of transponders also includes eliminating location redundant transponders that converge to the same point in space.

Results and Discussion

Testing confirmed that, with the right parameter settings and an initial transponder configuration, the positions of transponders can be adjusted and unnecessary transponders suppressed for effective recognition of a set of input patterns. An example of one of the learned configurations is shown in Figure 4. The white dots represent the input transponders (capitalized labels represent the "on" segments, lower case represent "off" segments), the bright green dots show the active hidden transponders, the bright red dots the active output transponders and the dark dots are suppressed transponders. Table 2 shows the settings of various parameters used in the example. Transponders were generated in a radial range of 5 to 150 as suggested by the modeling above. Limiting the density based on $\tau_{synapse}$, $\vartheta_{construction}$ and $n_{wavefronts}$ eliminated location redundancy while providing the necessary intersection sensitivity. The overall distribution of hidden and output transponders is set by the P_{radial}, D_{radial} and $D_{angular}$ (dilation) parameters. The effective density distributions for transponders are somewhat larger than the minimum detection probability space (radial power and dilation values < 1) because the position learning algorithm allows transponders to move across a wider area. The wavefront filtering mechanism was effective in allowing transponders to activate with the right

wavefront stimulus and avoid chaotic activations. Potentiation decay was highly effective in suppressing unused transponders with little impact on the overall behavior of the system. Synaptic fatigue was effective at identifying and suppressing non-discriminating output transponders but also may be a major contributor to the complexity of the solution space and may contribute to the failure to find feasible solutions in some configurations.

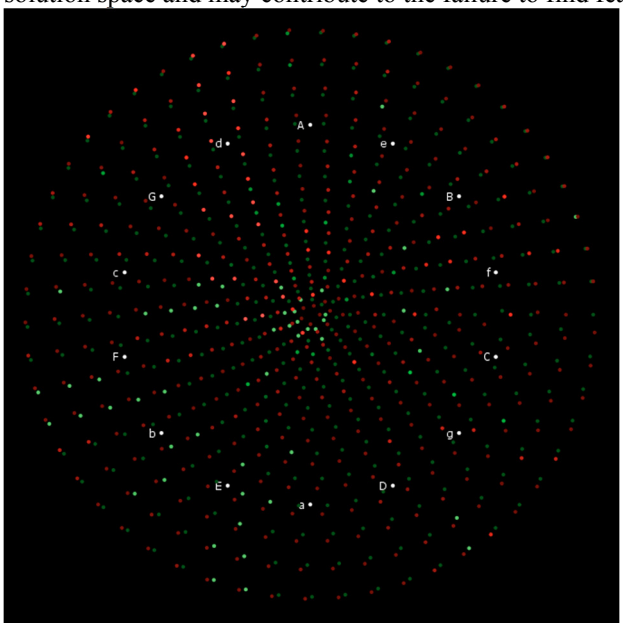


Figure 4 – Example Learned Topology for 7-Segment Digit Recognition

Table 1 shows the overall results of the configuration learning process in terms of transponder counts. The number of active hidden transponders is much higher than output transponders. For pattern recognition to be effective it is important to have large numbers hidden transponders to represent possible feature subsets. There are a large number of possible output transponders but only a subset is needed (ideally, one per input pattern of interest). The resulting overall number of active transponders (197 hidden and 34 output) is reasonable compared to previous results [9] (76 hidden and 10 output) where transponder positions were precomputed and the results are close to the theoretical minimum.

Analysis showed that a number of feasible solutions (parameter settings) exist but the number of parameters involved make the solutions difficult to find. Solutions were found using NetLogo's Behavior space capability to run multiple possible parameter combinations. The approach was to use baseline settings that had worked or come close in past attempts then vary radius power, radius dilation, angle dilation, and t-synapse to attain the best results. Modifying t-activation-charge was then used to fine tune the results. Gradient decent approaches were employed using the mean count of non-specific output transponders as the objective function combined with the evaluation criterial discussed above. The large number of parameters, discontinuities due to the activation functions, nonlinear relationships between many of the parameters and temporal relationships due to the iterative nature of the algorithms result in a very complex solution space shape that makes it difficult to apply gradient methods effectively. Viable solutions tend to occur in steep minima which are easily invalidated by small parameter changes. Solution space characteristics are partly due to the complexity of the model but many of the key parameters are directly related to the generation of the initial transponder field suggesting improvements in the initial field generation approach could make solutions easier to find.

The model exhibits many of the qualities of a complex adaptive system. It is a multi-agent adaptive system that self-organizes to recognize input patterns but it also displays some emergent behaviors that can make it difficult to analyze. Input transponders are predefined and produce predictable output. Hidden transponders adapt to the input wavefronts producing varying behavior as they converge but tend to stabilize over time and training samples. Potentiation decay and synaptic fatigue have complex interactions with the position learning process causing transponders to go in and out of activation ranges over time.

Table 2 - Example Parameter Values

Parameter	Value
$r_{min\text{-}hidden}$	8
$r_{max\text{-}hidden}$	150
$P_{radial\text{-}hidden}$	0.38
$D_{radial\text{-}hidden}$	0.69
$D_{angular\text{-}hidden}$	1
$r_{min\text{-}output}$	5
$r_{max\text{-}hidden}$	150
$P_{radial\text{-}output}$	0.32
$D_{radial\text{-}output}$	0.731
$D_{angular\text{-}output}$	0.907
A_{+}	1.5
A_{gain}	1
$Cost_{Activation}$	0.1
$\vartheta_{membrane}$	2.23
$\tau_{synapse\text{-}output}$	2.915
τ_{charge}	3600
$\tau_{synapse\text{-}hidden}$	2.96
τ_{gain}	12000
τ_{+}	0.3
$n_{wavefronts}$	3
$\vartheta_{construction}$	2.02
β	0.999998
$t_{training}$	100000

Table 1 - Configuration Summary Statistics

Transponder Statistic	Count	%
Input	14	
Hidden		
Active	197	49%
Suppressed	202	51%
Total	399	
Output		
Active	34	8%
Suppressed	373	92%
Total	407	
Total Active	245	30%
Total Initial	820	

4. Summary and Directions for Future Work

The initial results of this work have shown that, with the proper selection of parameter values and adaptive behaviors, an initial field of PWC transponders can self-organize into a configuration that recognizes input patterns when provided with a set of training data. The methods discussed allow the size of the initial field to be controlled and provide for dynamic suppression of unneeded transponders based on the patterns learned.

The recognition capabilities of the configurations produced are highly dependent on the behavioral parameters used by the process. While it has been seen that there are many possible combinations of parameter values that produce acceptable solutions, they are difficult to find given the large number of variables involved and the complexity of the solution space. Although the basic relationships between parameter values and transponder behavior have been defined, further research on the effects of parameter settings on the overall learning process and recognition outcomes is needed. Experiments to determine the effects of different training scenarios and the behavior of the resulting configurations with untrained examples are also needed to further understand the resiliency and limitations of the approach. The research also suggests that improvement in the initial field generation and the approach to suppressing non-specific transponders may provide further opportunities for improvement.

References

[1] Hinton, G., Deng, L. and Yu, D. (2012) "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups." IEEE Signal Processing Magazine 29 (6): 82-97.

[2] Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012) "ImageNet Classification with Deep Convolutional Neural Networks." Neural Information Processing Systems 30: 1097-1105.

[3] Izhikevich, E.M. and Hoppensteadt, F. (2009) "Polychronous Wavefront Computations." *International Journal of Bifurcation and Chaos* **19** (5): 1733-1739. http://www.izhikevich.org/publications/polychronous_wavefront_computations.htm

[4] Izhikevich, E.M. (2006) "Polychronization: Computation with spikes." *Neural Computing*. **18**, 245–282.

[5] Thomas, J. (2015) "The Geometry of Polychronous Wavefront Computation." ResearchGate. https://www.researchgate.net/publication/281156794_The_Geometry_of_Polychronous_Wavefront_Computation

[6] Thomas, J. A (2014) "Mathematical Treatise on Polychronous Wavefront Computation and its Application into Modeling Neurosensory Systems." ResearchGate.

[7] Hart, C. (2014) "Towards a Compiler for a Polychronous Wavefront Computer: Programming by Optimization". *Complex Adaptive Systems, Procedia Computer Science* **36**: 387-392. https://www.sciencedirect.com/science/article/pii/S1877050914012599

[8] Highland, F. and Hart, C. (2015) "Adaptation of Spike-Timing-Dependent Plasticity to Unsupervised Learning for Polychronous Wavefront Computing." *Complex Adaptive Systems, Procedia Computer Science* **61**: 314-321. https://www.sciencedirect.com/science/article/pii/S1877050915029762

[9] Highland, F. and Hart, C. (2016) "Implementing Multilayer Neural Network Behavior Using Polychronous Wavefront Computation." *Complex Adaptive Systems, Procedia Computer Science* **95**: 159-167. https://www.sciencedirect.com/science/article/pii/S1877050916324802.

[10] Highland, F. and Hart, C. (2016) "Unsupervised Learning of Patterns Using Multilayer Reverberating Configurations of Polychronous Wavefront Computation". *Complex Adaptive Systems, Procedia Computer Science* **96**: 175-184. https://www.sciencedirect.com/science/article/pii/S1877050916324838.

[11] Rosenblatt, F. (1961) *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms.* Spartan Books, Washington DC.

[12] Song, S., Miller, K. D., & Abbott, L. F. (2000) "Competitive Hebbian Learning Through Spike-timing-dependent Synaptic Plasticity." *Nature Neuroscience* **3**: 919-926. http://neurotheory.columbia.edu/~larry/SongNatNeuro00.pdf

[13] Gerstner, W. and Kistler, W. (2002) *Spiking Neuron Models. Single Neurons, Populations, Plasticity.* Cambridge University Press.

[14] Stein, R. B. (1967) "Some Models of Neuronal Variability." *Biophysical Journal* **7** (1): 37-68.

[15] Miller, A and Jin, Dezhe Z. (2013) "Potentiation decay of synapses and length distributions of synfire chains self-organized in recurrent neural networks." *Physical Review* E **88**, 062716.

[16] Simons-Weidemaier, N. S., Weber, M., Plappert, C. S., Pilz, P. K. D., Schmid, S. (2006). "Synaptic depression and short term habitation are located in the sensory part of the mammalian startle pathway." *BMC Neuroscience* **7**: 38-38

[17] Wilensky, U. (1999) NetLogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. http://ccl.northwestern.edu/netlogo

[18] Wood, F. W. (1910) U.S. Patent 974,943. U.S. Patent and Trademark Office.