

A study of the plastic Kuramoto model and its capacity to simulate mental synthesis

by

SPHESIHLE IMMANUEL MAVUSO

submitted in accordance with the requirements
for the degree of

MASTER OF SCIENCE

in the subject

PHYSICS

at the

UNIVERSITY OF SOUTH AFRICA

SUPERVISOR: Professor A. E. Botha

February 2026

Declaration

Name: Sphehile Immanuel Mavuso

Student Number: 21005060

Degree: Master of Science (Physics)

I declare that the dissertation “A study of the plastic Kuramoto model and its capacity to simulate mental synthesis” is my own work and that all the sources that I have used or quoted have been indicated and acknowledged by means of complete references and that any use of Artificial Intelligence (AI) has been fully disclosed.

I further declare that I submitted the dissertation to the appropriate originality detection system which is endorsed by Unisa and that it falls within the accepted requirements for originality.

I further declare that I have not previously submitted this work, or part of it, for examination at Unisa for another qualification or at any other higher education institution.

I further declare that where Artificial Intelligence (AI) tools have been used in the preparation of this thesis/dissertation, their use has been limited to ethical permissible support, has been fully disclosed, and does not replace my own original research, my independent critical thinking and analysis, or authorship responsibilities.

I understand that failure to disclose AI use, plagiarism and/or lack of academic integrity may constitute academic misconduct under Unisa’s policies.



SIGNATURE

21 May 2026

DATE

Dedication

This dissertation is dedicated to my partner, family, and supervisor, who supported me through the lowest of lows and the highest of highs in completing this work.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Prof. A. E. Botha, for his guidance, insight, and continued support throughout this research.

I am deeply grateful to my partner, Thandeka Masimula, and my family for their unwavering support, patience, and encouragement during this journey. Their belief in me made this achievement possible.

I also wish to acknowledge my good friend, Tshepo Nkambule, who followed my progress from the outset and provided valuable discussions, as well as assistance with implementation and debugging.

Finally, I would like to acknowledge the UNISA Department of Physics for partially funding my studies.

Abstract

Mental synthesis is the human ability to purposefully form novel mental imagery by combining memories of different modalities. The neuronal mechanisms that facilitate this process remain poorly understood. Mental synthesis is hypothesised to occur when memory encoding neuronal ensembles are synchronised while retaining aspects of the memories they encode. Presently, there is no mathematical model of such a process. In this study, we develop such a model by extending a plastic Kuramoto model with coupling that evolves via Hebbian learning. We construct neuronal ensembles with sparse random coupling and investigate the synchronisation of two such ensembles. Numerical simulations show that random coupling extends the parameter space supporting memory-encoding patterns and allows ensembles to synchronise while retaining features of their representations. This result suggests that memory encoding ensembles can be synchronised while retaining features of the memories they encode, representing the first step toward a mathematical model of mental synthesis.

Key terms: Neurological modelling, Mental synthesis, Neuronal ensembles, Plastic Kuramoto model, Hebbian learning, Spike-timing-dependent plasticity

List of Abbreviations

AP	-	Action potential
EPSP	-	Excitatory postsynaptic potential
LPFC	-	Lateral pre-frontal cortex
LTP	-	Long-term potentiation
LTD	-	Long-term depression
MVPC	-	Multivariate pattern classification
PFS	-	Prefrontal synthesis
STDP	-	Spike-time dependent plasticity

Contents

1	Introduction and Literature Review	2
1.1	Literature Review of Mental Synthesis	3
1.1.1	What is Mental Synthesis?	3
1.1.2	A Brief History of Mental Synthesis	4
1.1.3	Mental Synthesis in Daily Life	6
1.1.4	Clinical Significance of Mental Synthesis	7
1.1.5	The Neuronal Basis of Mental Synthesis	8
1.1.6	Lack of a Mathematical Models of Mental Synthesis	10
1.2	Literature Review of Suitable Mathematical Models of Mental Synthesis	10
1.2.1	Spike-Time Dependent Plasticity and Memory Encoding Neu- ronal Ensembles	11
1.2.2	Phase reduction and Kuramoto-type Models	14
1.2.3	Models that Combine Spike-time Dependent Plasticity and Kuramoto- type Models	18
1.2.4	Models of Multiple Neuronal Ensembles	22
1.3	Problem Statement and Methodology	24
1.3.1	Problem Statement	24
1.3.2	Methodology	26
2	Mathematical Details of the Model	27
2.1	Phase Reduction Formulation of STDP	27
2.2	Kuramoto Model	28
2.3	Kuramoto–Sakaguchi Model	30
2.4	Kuramoto–Sakaguchi Model with STDP	31
2.5	Modification of the Coupling Structure	32
2.6	Coupled Neuronal Ensembles	36
3	Code Development and Preliminary Benchmarking Simulations	39
3.1	Developing a Python Function for Simulating the Kuramoto Model . .	40

3.1.1	Direct Simulation of the Kuramoto Model	40
3.1.2	Numerical Integration Details and Order Parameter	41
3.1.3	Onset of Synchrony Validation	42
3.1.4	Computational Cost of the Direct Implementation	44
3.1.5	Introducing Just-In-Time Compilation	45
3.1.6	Large System Onset of Synchrony Experiment with Numba JIT	48
3.2	Developing a Python Function for Simulating the Kuramoto–Sakaguchi Model	50
3.3	Developing a Python Function for Simulating the Kuramoto–Sakaguchi Model with STDP	52
3.4	Developing Python Functions for Simulating the Kuramoto–Sakaguchi Model with STDP and Cyclic Coupling	55
3.5	Developing Python Functions for Simulating the Kuramoto–Sakaguchi Model with STDP and Random Coupling	62
3.6	Developing Python Functions for the Two-Ensemble Kuramoto–Sakaguchi Model with STDP and Random Coupling	64
4	Effects of Random Coupling	72
4.1	Phase Synchronisation and the Emergence of a Pacemaker	72
4.2	Effect of Random Coupling on Chimera States	75
4.3	Effect of Random Coupling on the Interval of Stable Chimera States . .	80
5	Simulations of Coupled Ensembles	89
5.1	Metrics Used	91
5.1.1	Global Synchronisation Metric	91
5.1.2	Phase Pattern Correlation Metric	92
5.2	Synchronisation of Two Ensembles	93
5.3	Retention of Memory Characteristics	98
6	Discussion and Conclusion	102
	Bibliography	105

Chapter 1

Introduction and Literature Review

The neuronal mechanisms of brain functions are fascinating and have long intrigued researchers. Understanding the neuronal mechanisms of brain functions promises to provide first principles insights into how the brain gives rise to the mind and sheds light on overcoming neurological pathologies. Hence, over the years, researchers have studied the neuronal basis of various brain functions, and some of this research has led to practical applications such as machine learning and artificial intelligence.

Mathematical models of the neuronal mechanisms of brain functions have been fruitful starting points for understanding the neuronal dynamics underlying brain function. Such models offer the advantage of simulating brain functions without the need for costly clinical and imaging experiments. They also offer the opportunity to develop our understanding of brain mechanics at the level of a single neuron. Although mathematical models have been developed for various brain functions, none have been reported for mental synthesis. This dissertation addresses this gap by developing a model based on phase reduction techniques and concepts from neuroscience. The developed model is simulated to test the leading hypothesis of the neuronal dynamics of mental synthesis.

In the following chapters we will discuss how the proposed model was developed and tested. In this first chapter, we define the concept of mental synthesis and briefly discuss its history, external manifestations, and clinical significance. We then present a review of the leading hypothesis of the neuronal dynamics behind mental synthesis. Next, we review some of the mathematical models that could be combined to model mental synthesis. This chapter concludes with a problem statement concerning the objectives and methodology of our research. Chapter 2 presents the mathematical details of the

proposed model, while Chapter 3 discusses the development and benchmarking of the computational script used to implement the model. Chapters 4 and 5 present the results obtained with the developed model. Finally, Chapter 6 concludes the dissertation by comparing its findings with previous research, discussing its limitations, and suggesting some directions for possible future research.

1.1 Literature Review of Mental Synthesis

This section reviews some of the literature related to mental synthesis and also acts as an introduction to the topic. The material is arranged into the following subsections: In Section 1.1.1, mental synthesis is defined. Section 1.1.2 discusses its history, while in Sections 1.1.3 and 1.1.4 we discuss the role of mental synthesis in daily life and clinical settings, respectively. Although the discussion in Sections 1.1.3 and 1.1.4 is not essential to follow the main thrust of the dissertation, it is provided to demonstrate the broader context of mental synthesis. In Section 1.1.5, we introduce the hypothesised neuronal mechanisms of mental synthesis, and finally, in Section 1.1.6, we stress the current lack of any mathematical models of mental synthesis.

1.1.1 What is Mental Synthesis?

According to Vyshedskiy [1], “Mental synthesis is the voluntary process of synthesising a new, never-before-seen image from two or more mental images”. It is the process that is thought to occur when one visualizes, say, a lion sitting on Bill Clinton’s lap. This never-before-seen image is formed by combining the memories of a lion and Bill Clinton into a new image that is not present in memory [2].

Mental synthesis need not be limited to mental images [1]. It can extend to novel mental experiences that involve other sensory modalities such as motion and scent [1]. For example, one can mentally experience a walk in a lavender garden on a warm summer morning without having experienced such a walk in the past. Mentally experiencing such a scene would not be limited to visual imagery, it would involve the mental experience of the motion of walking along with the sensations of heat and lavender scent. Hence, the “image” in the definition of mental synthesis can refer to sensory experiences beyond visual images. It can refer to sensations such as sound, scent, taste, motion, and temperature. Indeed, Vyshedskiy states that the definition of mental synthesis can

be extended to “... the process of creating any new mental experience by combining any number of previous experiences from memory” [1].

Mental synthesis does not encompass imagination, dreaming, and mental imagery [1]. Although these functions involve mental experiences from memory, they differ from mental synthesis because they do not require the creation of new images and can occur subconsciously and involuntarily [1]. For example, imagination and mental imagery can involve simple recall, where no new mental experience is formed [2], and may also occur subconsciously and involuntarily, as in spontaneous flashbacks [3].

In summary, mental synthesis is the process of combining mental experiences from memory to form novel, never-before experienced mental scenes. These scenes can be unimodal, as in creating a new image, or multi-modal, as when creating a scene with multiple sensory modalities. Finally, mental synthesis is voluntary. This requirement excludes processes in which novel mental experiences occur involuntarily, such as dreaming or hallucinating.

1.1.2 A Brief History of Mental Synthesis

The term mental synthesis is borrowed from psychology research [1]. It was first introduced by Finke et. al. [4] in 1988 and understood to mean the ability to identify novel images from mental combinations of symbols and geometric shapes [5–7]. This research was geared towards understanding how humans form never-before-seen mental images from mental images of known objects. Reviewing this research sheds light on how mental synthesis has been understood in the past and how it has been investigated in behavioural studies.

Finke et al. [4] developed the first experimental test of mental synthesis. They verbally presented symbols and shapes to participants and instructed them to hold the presented items in working memory. The participants were then allowed to freely combine the symbols and shapes or reorient them to identify novel combinations. The experimenters assessed the identified combinations and rated their creativity. The recognisability and creativity scores of the identified combinations served as a measure of an individual’s ability to perform mental synthesis [4].

Studies designed around Finke et al.’s test dominated mental synthesis research from the 1990s to the early 2010s [5, 7–10]. Using the test, researchers investigated mental synthesis ability in individuals of different ages and sensory and cognitive profiles [7, 8,

10], and when the synthesised forms were presented in different modalities [5].

Mental synthesis research has shown that the ability to perform mental synthesis is well-developed by the age of 6 [8, 10]. However, children around this age struggle to form complex mental composites, suggesting that the ability to perform mental synthesis depends on the efficiency of control processes, which improves with age [8]. Mental synthesis ability among young children has also been shown to correlate with better performance in tasks that measure elaboration ability, thought flexibility, and thought originality [10].

Mental synthesis has also been shown to be present in blind individuals. Eardly and Pring [7] presented 2D and 3D symbols to blind individuals and individuals with sight. The participants synthesised these forms by reorienting or combining them. The authors found that the mental synthesis performance of blind participants was worse for 2D objects but comparable to that of seeing participants when the synthesised symbols were presented as 3D objects. The improved mental synthesis performance for 3D objects among blind participants was attributed to their greater familiarity with 3D objects compared to 2D objects [7].

The modality in which synthesised symbols and forms are presented has been shown to affect mental synthesis performance [5, 7, 9]. In 2003, Pearson et al. [5] showed that mental synthesis performance was better under visual presentation than under verbal presentation. These results are thought to indicate that visual presentation enables the use of short-term memory instead of long-term memory, improving mental synthesis ability [5].

The above research demonstrates that mental synthesis develops early in human development but advances with the development of other executive functions [8, 10]. Further, better mental synthesis ability is correlated with improved abilities in other cognitive functions [8]. The research shows that mental synthesis is present in blind individuals [7] and can therefore be reasonably extended to individuals with other sensory challenges. Finally, the effect of presentation modality on mental synthesis performance indicates that memory plays an essential role in mental synthesis. That is, reducing cognitive load by making the synthesised forms available in short-term memory enhances performance, while relying solely on long-term memory increases cognitive load and reduces performance [5].

1.1.3 Mental Synthesis in Daily Life

Mental synthesis manifests in various forms in daily life. It manifests in the structure of human language, as well as in creativity, innovation, and behaviour [1]. A reduced ability to perform mental synthesis, either as a result of atypical brain development or the lack of social interaction at an early age, can lead to communication and problem-solving difficulties in adulthood [11–15]. Thus, understanding mental synthesis can play a crucial role in developing diagnostic and therapeutic approaches for overcoming mental synthesis limitations.

Mental synthesis is used when communicating novel imagery [1, 12, 13, 16]. The speaker constructs the intended imagery through mental synthesis, while listeners rely on the same process for comprehension. It also facilitates story-telling by enabling audiences to conjure previously unseen mental images [1]. Furthermore, mental synthesis is essential in active teaching, where learners must construct mental representations of information communicated by a teacher to improve understanding [1].

Mental synthesis is used in creativity and innovation [1, 2, 17]. Artists use mental synthesis to combine images from their memory into novel forms. Engineers use mental synthesis to create blueprints of new objects through the mental combination of parts [2]. Finally, scientists employ mental synthesis when they combine known information to construct novel models of phenomena that may be invisible to the eye [1].

Some human behavioural traits heavily involve mental synthesis [1, 12]. For example, during pretend play, children use mental synthesis to combine their memories to form new mental experiences [1]. Mental synthesis is also employed in strategy games such as chess and Go, where current scenarios and experience are combined to mentally simulate the impact of different moves [1].

The development of recursive language is arguably the most important external manifestation of mental synthesis. All human language, including sign language, has a recursive form [18–20]. Recursive structure allows humans to understand the difference between "My friend ate a fish" and "A fish ate my friend." To grasp the meaning of these sentences, mental synthesis is used to combine the mentioned objects and create a mental image of the described scenarios and extract meaning [1, 2].

A lack of exposure to recursive language can have adverse effects on childhood development [13, 15, 17]. The ability to perform mental synthesis develops in a strong critical period during childhood, and individuals who are not exposed to recursive

language in this period never develop the ability to perform mental synthesis [13, 17].

Research suggests that the capacity for mental synthesis may have contributed to the evolution of recursive language in humans [1, 17]. Although the physiological apparatus of modern speech evolved 600 000 years ago among hominids, indicators of modern imagination, used as evidence for the development of mental synthesis, only began to appear in the archaeological record around 70 000 to 100 000 years ago [17, 21]. This delay between the emergence of the speech apparatus and the later appearance of representational art suggests that the cognitive ability for mental synthesis developed later than the physiological capacity for speech. One explanation for this delay is that mental synthesis can only be acquired during a limited developmental window in early childhood [17].

A recent evolutionary mathematical model proposes that a genetic mutation extending this developmental window allowed children to acquire mental synthesis and to develop recursive language through communication with one another [17]. According to this model, the interaction of children with this extended developmental period may have transformed non-recursive communication systems into recursive language. The model further suggests that this transition occurred around 70 000 years ago, coinciding with the earliest archaeological evidence of modern imagination, such as representational art [17].

1.1.4 Clinical Significance of Mental Synthesis

The main clinical significance of mental synthesis is its relation to autism spectrum disorder (ASD). Individuals with ASD often display a symptom called stimulus over-sensitivity, where they cannot mentally combine separate object [12]. This symptom manifests as a difficulty to understand spatial prepositions, semantically-reversible sentences, and recursive sentences [12, 15, 16, 22].

Individuals with this symptom may struggle to understand instructions such as “pick up the red crayon that is under the table.” The individual would fail to mentally synthesise the crayon with its colour and location, and may attend to the word “crayon” and ignore its location and colour. Hence, they may pick up any available crayon [12]. This symptom is observed in 30 to 40% of children with ASD and could be as high as 60% among children enrolled in special ASD schools [12, 15, 22].

Stimulus over-sensitivity points at a weak mental synthesis ability among chil-

dren with ASD. Research has shown that children with ASD may acquire vocabulary without overcoming stimulus over-sensitivity, suggesting a weak acquisition of mental synthesis [12, 15, 16, 22]. For these individuals, common intelligence tests based on assessing vocabulary may not detect mental synthesis deficits [12, 15, 22]. Hence, tests and therapeutic interventions that employ the understanding of mental synthesis have been developed [12, 15]. These interventions allow children with ASD to practice combining mental images presented verbally and non-verbally to develop their mental synthesis ability and alleviate stimulus over-sensitivity symptoms [12, 15]. These interventions are showing promising pre-randomized study results, demonstrating significant language acquisition improvements among children with ASD [15].

1.1.5 The Neuronal Basis of Mental Synthesis

The neuronal mechanisms underlying mental synthesis are not well understood. Currently, the only objective study on the neuronal mechanisms of mental synthesis was published by Yogimida et al. [23] who used fMRI to measure the brain activity of individuals performing mental synthesis tasks. They found that performing mental synthesis tasks is correlated with activity in the inferior frontal and inferior temporal cortices of the left hemisphere. They concluded that these networks form part of a brain network that facilitates mental synthesis [23].

The current leading hypothesis of mental synthesis suggests that mental synthesis occurs when the lateral pre-frontal cortex (LPFC) synchronizes two or more memories stored in an ensemble of neurons with synchronous activity and a defined synaptic map [1, 2]. The synchronous activity of these ensembles creates the novel mental experience observed during mental synthesis [1, 2]. This hypothesis has not been tested experimentally or computationally. However, the centrality of the LPFC in enabling mental synthesis has been confirmed by a mental synthesis deficit among patients with damage to the LPFC [24]. Further, neuroimaging studies have shown that the LPFC is inactive during sleep and dreaming [25], suggesting that the LPFC is only essential when consciously combining mental images. Because of this hypothesis, mental synthesis is also called prefrontal synthesis (PFS) [15–17].

The hypothesis that mental synthesis occurs when neuronal ensembles that store memories are synchronized stems from observations that perception is facilitated by the synchronisation of neurons that are activated by external stimuli [1, 26–30]. For example, in an experiment with human subjects performing challenging perceptual tasks,

inter-cranial recordings showed that perception could be predicted by synchronous activity in the occipitotemporal cortex, the prefrontal cortex, and the hippocampus, while non-recognition was associated with drastically reduced synchronisation between these areas [31].

Experiments indicate that memory recall reinstates the same neuronal ensembles engaged during perception. At the single-neuron level, recordings in the human temporal lobe, hippocampus, and entorhinal cortex show that neurons activated while viewing a cinematic episode or audiovisual sequence fire when that episode is later recalled [32, 33]. At a distributed level, fMRI studies reveal that ventral-temporal cortex regions engaged during the perception of faces, buildings, tools, or food are reactivated during imagination or recall, with particularly strong overlap for famous people and buildings [34].

Complementary studies using fMRI and multivariate pattern classification (MVPC) demonstrate that recalling or visualizing objects elicits neural activity patterns in the visual cortex that closely resemble those observed during actual perception [35, 36]. For instance, Cichy et al. [35] showed that category-specific patterns observed during recall could predict patterns observed when participants viewed the same object. Klein et al. [36] found that imagining the reorientation of a bow tie activated visual areas in a manner that matched the perception of the same reorientation. These findings indicate that perceiving and recalling objects largely rely on overlapping neural representations.

Synchronisation has been shown to support recall [37–39]. Intra-cranial EEG studies have shown that the successful retrieval of memory is associated with synchronous activity in the hippocampus, temporal lobe, and prefrontal cortex. This synchronous activity supports the formation of networks that selectively reassemble stored representations [37]. Other studies have shown that synchrony in the hippocampus during memory retrieval organises spike timing to reactivate memory encoding ensembles [38, 39]. These findings suggest that synchronous activity facilitates recall through coordinating the reinstatement of neuronal ensembles.

Observations that similar neurons and neuronal ensembles are active when objects are observed and recalled suggest that memory is encoded in an ensemble of neurons with synchronous. In addition, the central role of synchronisation in both recall and perception suggests that it could play a role in mental synthesis, particularly during the activation of memories that make up the synthesised experience. Finally, the lack of mental synthesis when the LPFC is damaged suggests that this structure orchestrates mental synthesis. These observations support the hypothesis that mental synthesis

arises when the LPFC coordinates the activation of memories by synchronising neuronal ensembles representing the components of the synthesised experience.

1.1.6 Lack of a Mathematical Models of Mental Synthesis

At present, the hypothesis of the neuronal mechanics of mental synthesis does not have an accompanying mathematical model. Further, although there has been an fMRI study on the neuronal activity associated with mental synthesis, no accompanying computational model [23], such as an MVPC model, has been developed for predictive studies.

The current understanding of the neuronal mechanisms of mental synthesis points at the importance of cognitive load and memory [5], the efficiency of control processes [8], orchestration by the LPFC [1, 2, 17], and neuronal synchronisation [1]. Further, the distribution of the neurons involved in perception and the overlap of these neurons with those activated during recall suggest that the neuronal ensembles involved in mental synthesis may have both long and short-distance connectivity.

1.2 Literature Review of Suitable Mathematical Models of Mental Synthesis

As discussed above, mental synthesis is hypothesised to occur when neuronal ensembles that encode different memories are synchronised [1]. Hence, a mathematical model of mental synthesis that follows from this hypothesis must model memory encoding neuronal ensembles as ensembles made up of neurons with synchronous activity. It should also model the synchronisation of two or more such ensembles. This section reviews literature on mathematical models that can be integrated to realise such a mathematical model.

Section 1.2.1 introduces spike time-dependent plasticity (STDP) and its role in the formation of memory encoding neuronal ensembles. Section 1.2.2 discusses phase reduction theory and Kuramoto-type models. It emphasizes the historical development of these models along with their generalizations for application in various physical contexts. Section 1.2.3 reviews studies that combine STDP-based models and Kuramoto-type models to model neuronal ensembles that encode memory. Finally, Section 1.2.4 provides a snapshot into recent models of two or more neuronal ensembles.

1.2.1 Spike-Time Dependent Plasticity and Memory Encoding Neuronal Ensembles

Memories are thought to be stored as the patterns of synaptic connections between neurons that are activated during perception and learning [40–42]. The mechanism behind the formation of these synaptic connections is synaptic plasticity, which is the change in the strength of synaptic connections based on the pattern of activity in the synapses [40]. Synaptic plasticity is based on the Hebbian theory that neurons that fire together wire together [43]. Although the idea that memories are stored as patterns of synaptic connections has been criticised, empirical findings continue to demonstrate its centrality in the neuronal basis of learning and memory [42].

In 1949, Hebb postulated that neuronal ensembles are formed when the connections between neurons that are repeatedly involved in each other’s firing are strengthened [43]. He suggested that if neuron A is repeatedly involved in the firing of neuron B, then the connection between neurons A and B is enhanced [43]. This enhancement creates a neuronal ensemble that facilitates perception, learning, and memory.

Hebb’s theory has been developed into a simple model of synaptic plasticity called spike-time dependent plasticity (STDP). According to STDP, when a post-synaptic neuron repeatedly fires after a pre-synaptic neuron, the strength of the synapse increases. In contrast, when the post-synaptic neuron repeatedly fires before the pre-synaptic neuron, the synaptic strength between the neurons decreases [44–46].

A synapse is strengthened or potentiated when the same stimulus from the pre-synaptic neuron produces a larger response in the postsynaptic neuron. Long-term potentiation (LTP) is when this effect lasts for hours to days [47]. However, if the same pre-synaptic stimulus produces a weaker response in the postsynaptic neuron, the synapse is weakened or depressed. When this effect lasts for hours and days, the synapse has undergone long-term depression (LTD) [47].

Memory storage requires both LTP and LTD [48]. On the one hand, LTP enhances signal transmission and amplification among connected neurons in a memory track. On the other hand, LTD eliminates connections that were not essential during encoding, making the transmission map more efficient [49]. Although LTP and LTD are due to a dynamic interplay between neurotransmitters, intracellular ions, and membrane gates, they can be effectively modelled using only the timing of action potentials in pre- and postsynaptic neurons through STDP [50].

STDP has been demonstrated empirically in vivo and in vitro [44–46]. In 1997, Markram et al. [46] performed dual-whole cell voltage recordings from neighbouring thick, tufted pyramidal neurons in layer 5 of the neocortex. They used the amplitude of the excitatory post-synaptic potential (EPSP) as a measure of the synaptic strength between pre-synaptic and post-synaptic neurons. They initiated a pre-synaptic AP that produced sub-threshold EPSPs in the post-synaptic neuron. They found that repeatedly initiating a post-synaptic AP 10 ms after an EPSP increased the amplitudes of subsequent EPSPs, while repeatedly initiating a post-synaptic AP 10 ms before an EPSP reduced the amplitude of subsequent EPSPs [46]. Identical experiments were performed with a delay of 100 ms and observed no EPSP effects, suggesting that EPSP amplitudes increase or decrease if the time delay between the presynaptic neuron and postsynaptic AP is less than 100 ms [46].

In 1998, Zhang et al. [45] performed in vivo whole-cell recordings from neurons in the optical tectum of young *Xenopus* tadpoles to monitor changes in retinotectal synaptic strengths following the repetitive electric stimulation of retinal neurons. They found that the amplitudes of subthreshold synaptic inputs that repeatedly occurred 20 ms before the peak of a postsynaptic spike increased. In contrast, the amplitudes of subthreshold inputs that occurred within 20 ms after the peak of a postsynaptic AP decreased [45]. In the same year, Bi et al. [44] induced persistent potentiation and depression in glutamatergic neurons of cultured disassociated rat hippocampal neurons. They found that repetitive postsynaptic spiking, initiated by a supra-threshold synaptic input or direct potentiation, within 20 ms of pre-synaptic activation resulted in synaptic potentiation. Further, postsynaptic spiking within 20 ms before pre-synaptic activation led to LTD [44].

The above empirical findings inspired the development of a mathematical model of STDP:

$$\Delta A = \begin{cases} A_+ e^{-\frac{t_{post}-t_{pre}}{\tau_p}}, & t_{post} > t_{pre}, \\ A_- e^{-\frac{t_{post}-t_{pre}}{\tau_d}}, & t_{post} < t_{pre}, \end{cases} \quad (1.1)$$

where ΔA is the percentage change in the post-synaptic response due to a pre-synaptic input and A_+ and A_- are the maximum potentiation and depression, respectively, with $A_- < 0$. t_{post} and t_{pre} are the times of the post-synaptic and presynaptic APs, respectively, and τ_p and τ_d are the maximum potentiation and depression time delays, respectively [51].

Fitting Equation 1.1 to different experiments produces different parameter values. A fit to the data of Bi and Poo [44] gives $A_+ = 100\%$, $A_- = -40\%$, $\tau_p = 100$ ms, and

$\tau_d = 60$ ms. In contrast, Zhang et al. [45] reported $A_+ = 80\%$, $A_- = -40\%$, $\tau_p = 50$ ms, and $\tau_d = 50$ ms. These differences suggest that different neuronal populations may give rise to different STDP parameters.

Experimental results have also shown that STDP varies across neuron types and depends on the presence or absence of NMDA receptors [44, 45]. Zhang et al. and Bi et al. found that the use of NMDA receptor inhibitors stopped both LTP and LTD for correlated neuronal spikes [44, 45]. Bi et al. [44] found that LTP and LTD did not occur in GABAergic postsynaptic neurons and that blocking L-type calcium channels prohibited LTD induction and reduced the extent of LTP. These findings indicate that STDP is not necessarily a uniform synaptic rule but instead emerges from a complex interplay of receptor dynamics, cell type-specific mechanisms, and intracellular signaling pathways [52].

Even though STDP arises from a complex interplay of receptor dynamics, cell type-specific mechanisms, and intracellular signalling pathways, the simple STDP rule introduced in Equation 1.1 and similar rules are effective in computational models. Early research demonstrated that minimal timing-dependent plasticity rules can produce the selective strengthening or weakening of synapses based on repeated temporal correlations in neuronal activity [51, 53]. These simple models have been shown to give rise to representational maps based on neuronal activity without requiring detailed biophysical specificity [50, 54]. Hence, simple STDP rules, like those in Equation 1.1, are tractable abstractions that capture the computational essence of synaptic plasticity.

This abstraction has been used to develop computational models of memory in networks that mimic neuronal networks. Networks with weights that vary via STDP and STDP-like rule have been shown to store stable patterns and associations, maintain correlations between input and output activity, and develop long-term synaptic configurations that are consistent with neuronal memory tracts [51, 55]. Further, pair-wise STDP has been used to produce self-sustained firing trajectories that resemble sequential memory in biological circuits [56]. These applications demonstrate that detailed molecular and cellular mechanisms need not be accounted for in capturing the computational principles underlying memory formation in neuronal networks.

This subsection introduces STDP along with its empirical evidence and computational applications. It demonstrates the efficacy of applying simple STDP rules that do not account for molecular and cellular complications in the modelling of neuronal networks of memory. This efficacy establishes the simple STDP rule introduced in this subsection as an effective mechanism for modelling the connectivity of memory encod-

ing neuronal ensembles. What remains is a model of neuronal synchronisation. Models describing synchronisation within and between neuronal assemblies are discussed next.

1.2.2 Phase reduction and Kuramoto-type Models

The synchronisation of groups of neurons has been modelled in using various biophysically plausible models. Certain modelling approaches employ conductance-based models such as the Hodgkin–Huxley or FitzHugh–Nagumo models, which describe the excitation dynamics of individual neuron membranes [57–59]. Coupling these neurons into networks revealed how excitatory and inhibitory synapses can generate stable phase relationships, synchronous firing, or cluster states depending on synaptic strengths, network topology, and the properties of the neurons [60, 61]. Simpler spiking neuron formulations, such as the integrate-and-fire and θ -neuron models, which model membrane excitability dynamics of individual neurons while enabling the analysis of network-level behaviour, have been developed [62]. These models reduce computational loads since they are less detailed. Both the conductance-based and simplified models produce collective neuronal activity from interactions of individual neurons.

For larger and more complex neuronal networks, models with individual neuron details become analytically intractable. This problem has motivated the application of phase-reduction approaches to the modelling of interacting neurons. In these approaches, neurons are modelled as phase oscillators with a phase that tracks the trajectory of a neuron’s spiking cycle during tonic spiking and a natural frequency that corresponds to the intrinsic frequency of this cycle [63]. This approach eliminates the need for the biophysical details of individual neurons, significantly simplifying the computation problem. It captures many qualitative features of neural synchronisation, including phase locking, clustering, and resonance, while abstracting detailed voltage and gating dynamics [64]. It therefore acts as a bridge between biophysical realism and tractable mathematical analysis.

The Kuramoto model is the paradigmatic model of synchronisation [64]. In this model, the coupling between phase oscillators has a sinusoidal relationship to their phase difference [65, 66]. The dynamics of the model are such that faster oscillators increase the speed of slower oscillators and vice versa [65]. For a sufficiently large coupling, the oscillators are spontaneously synchronised and entrained to the same frequency. The oscillators can be fully synchronised, where they have the same frequency and phase, or phase synchronised, where they have the same frequency but have constant

phase differences [67]. For intermediate coupling, some oscillators can be entrained synchronised while others remain incoherent [64]. This state is known as a chimera state [68, 69].

The Kuramoto model is based on ideas that date back to Huygens in 1665, who discovered that pendulum clocks connected by a beam eventually synchronised their motion [70]. This synchronisation has been observed in fireflies [71], the radio frequencies of electrically connected generators [72, 73], and cardiac pacemaker cells [74].

Early studies of synchronisation focused on the dynamics of limit cycles of the objects in the synchronising population [75]. A limit cycle is a closed trajectory in the phase space of a dynamical system, corresponding to persistent periodic behaviour [72]. For example, the oscillations of a clock or the tonic spiking of neurons can be described in terms of limit cycles [75]. However, this approach has limitations. Modelling the full dynamics of limit cycles often requires a high-dimensional representation of a system. This can make analysis and simulations computationally expensive and conceptually challenging [61]. Furthermore, it may not adequately describe the essential features of synchronisation, particularly for systems with weak coupling or heterogeneity [70]. These limitations are pronounced in complex systems like biological networks, where intrinsic noise and variability play a significant role [61, 70].

Although the study of limit cycles remains an active area of research, with developments such as the analysis of synchronisation patterns in large-scale neuronal networks and the advancement of computational tools for high-dimensional dynamical systems [61], alternative approaches have also been developed [70]. One such approach is phase reduction theory, which dates back to Malkin’s work in the 1950s [76]. Phase reduction simplifies the analysis of oscillatory systems by reducing the dynamics to a single variable, the oscillator phase [70]. Rather than analysing the full limit cycle, this framework focuses on the evolution of the phase and its interactions with other oscillators [70]. This framework was further developed by Winfree, who applied it to biological systems such as circadian rhythms [77], and by Kuramoto and Sakaguchi, who used it to study dissipative and diffusive chemical systems [78, 79]. Kuramoto’s work led to the Kuramoto model, which describes the emergence of synchronisation in populations of coupled oscillators [78]. The model has since been extended to study synchronisation phenomena across a wide range of fields, including neuroscience, physics, and the social sciences [64].

The Kuramoto model is defined as

$$\dot{\theta}_i = \omega_i + \frac{K}{N} \sum_{j=1}^N \sin(\theta_j - \theta_i), \quad (1.2)$$

where ω_i is the natural frequency of the i^{th} oscillator, K is the weak coupling between oscillators, θ_i is the phase of each oscillator, and N is the number of oscillators. The natural frequencies follow a random distribution $g(\omega)$ [65] that is often assumed to be unimodal and symmetric about the origin [80].

In the Kuramoto model, synchronisation is measured using the order parameter, which can be obtained by rewriting the Kuramoto model in complex form [64]:

$$r \exp(i\psi) = \frac{1}{N} \sum_{j=1}^N \exp(i\theta_j), \quad (1.3)$$

where r is the order parameter and ψ is the average phase of the oscillators.

The order parameter measures synchronisation in a scale from zero to one [65]. When the order parameter is zero, none of the oscillators are synchronised, and when it is one, all the oscillators are synchronised. Values between zero and one indicate the coexistence of one or more clusters of synchronised oscillators with incoherent oscillators [80].

Since the order parameter is used as a measure of synchronisation, it can be used to track transitions in and out of synchronisation. It has been observed that when the coupling K is increased from zero to some critical coupling K_c , the order parameter r remains zero until K_c , where it leaves zero and increases continuously with K tending towards one [64]. The critical coupling K_c can be calculated analytically assuming $N \rightarrow \infty$ and the natural frequencies follow a random distribution $g(\omega)$ that is unimodal and centred at $\omega = 0$ [64, 78]. Under these assumptions,

$$K_C = \frac{2}{\pi g(0)}. \quad (1.4)$$

This critical coupling can be confirmed in simulations with finite N , although with finite-size effects that broaden or slightly shift K_c [70].

Various parameters can be added to the Kuramoto model to represent the properties of different systems [64]. The natural frequencies can be taken from various distributions representing the natural frequency distributions of the studied oscillators [81]. In neuroscience, natural frequencies are the frequencies of the limit cycle of spiking

neurons. They can follow Gaussian and log-normal distributions, representing intrinsic variability and skewed population firing rates [26, 61].

A frustration or phase lag term can be added to the phase difference in the sinusoidal function [64]. When this term is added, the model is called the Kuramoto–Sakaguchi model and becomes

$$\dot{\theta}_i = \omega_i + \frac{K}{N} \sum_{j=1}^N \sin(\theta_j - \theta_i - \alpha), \quad (1.5)$$

where α is the frustration term.

The term was first introduced by Sakaguchi in 1986 with the restriction that $|\alpha| < \pi/2$ [82]. Sakaguchi noted that introducing this term with the stated restriction shifted the critical coupling of the system. He found that the critical coupling becomes

$$K_c = \frac{2}{[\pi g(\omega) \cos \alpha]}. \quad (1.6)$$

Subsequent research has shown that for $|\alpha| > \pi/2$, $\cos \alpha \leq 0$ and frequency synchronisation becomes impossible [64]. In this frustration region, the system may exhibit complex synchronisation states such as chimera states [64]. The frustration term captures competition among oscillators in a network [70]. In neuronal systems, it represents delays or asymmetries in signal transmission between neurons [83].

Along with the frustration parameter, other extensions to the Kuramoto model have been explored but are not central to the present study. An inertia term has been introduced to the Kuramoto model, turning it from a first-order equation to a second-order equation. The inclusion of an inertia term accounts for momentum in oscillator systems and turns the system from over-damped to under-damped. Introducing an inertia term delays K_C or sharpens it. Introducing inertia is essential for systems with momentum, such as power grids [84, 85].

Models with stochastic noise have also been introduced [86]. These models account for intrinsic fluctuations in the dynamics of oscillators in a system. Noise extends the incoherent states and increases critical coupling. However, in some regions, it stabilizes partial synchrony or induces synchronisation through stochastic resonance effects.

Meanwhile, the coupling function, $\sin(\theta_j - \theta_i)$, can be changed to capture different interaction dynamics between oscillators. Some studies use the cosine function instead of the sine function to the coupling function [63]. Using the cosine function

captures systems with different coupling symmetries or periodicities [63]. More complex coupling functions have been used in neuroscience to represent nonlinear response properties, multiple firing thresholds, and adaptive coupling dynamics [63]. The coupling function can also consider higher-order terms. Higher-order terms are terms such as $\sin(2(\theta_j - \theta_i))$, and using them stems from Taylor expansion, suggesting that the original form of the Kuramoto model can be considered to be an approximation where the phase difference is dominant over other terms [64]. Using higher-order terms enables the consideration of triplet coupling and other forms of coupling, such as time-delayed coupling, nonlinear coupling, and frequency-modulated coupling [87, 88]. In neuroscience, higher-order terms have been used to represent network resonance phenomena, multiplexed information encoding, heterogeneous coupling delays, and cross-frequency coupling [89, 90]. In other fields, they have represented complex synchronisation in power grids, nonlinear interactions in ecological systems, phase locking in lasers, and emergent behaviour in social systems [72, 91].

Although including inertia and stochastic noise and changing the coupling function demonstrates the versatility of the Kuramoto model, these changes are not analysed in significant detail because including them would go beyond developing a simple model of mental synthesis. These parameters and changes to the Kuramoto model are only discussed for completeness.

The Kuramoto model is a good candidate for modelling synchronisation in neuronal systems. It has been used in various studies for this purpose, with complicated conductive models of neurons and simple phase reduction approaches. However, without adaptive coupling, the resulting ensembles do not resemble memory storing neuronal tracts, since memory storage equally depends on neuronal synchronisation and the resulting coupling map. Hence, in the next subsection, we discuss Kuramoto models with adaptive coupling and their application in neuronal models of memory.

1.2.3 Models that Combine Spike-time Dependent Plasticity and Kuramoto-type Models

Standard Kuramoto-type models, such as Equations 1.2 and 1.5, have global coupling, where each oscillator is coupled to every oscillator with a constant coupling strength that does not vary with time [92]. Variations that adopt coupling strengths that vary via various learning rules exist but are generalisations [92, 93]. Introducing varying coupling produces richer, more complex, and dynamical behaviour such as multi-stability [94]

and cluster states [95]. Furthermore, abandoning global coupling for more realistic network topology, such as sparse coupling, produces emergent properties such as chimera states [47, 96, 97], which are observed in real-world networks. This Section reviews literature on Kuramoto-type models with coupling that varies via STDP and STDP-type learning rules with global and sparse coupling. It seeks to demonstrate that models integrating STDP and STDP-type learning rules with Kuramoto-type models produce candidate models of memory encoding neuronal ensembles.

STDP and STDP-type learning rules, which depend only on spike-timing to update synaptic strengths, are ideal learning rule candidates for Kuramoto-type models. The pre-synaptic and post-synaptic times used in these learning rules can be replaced with a phase that is defined for the oscillators modelled in Kuramoto-type models. For the Kuramoto and related models, the coupling is changed to A_{ij} , where i and j are the indices of the presynaptic and postsynaptic oscillators, with A_{ij} representing the coupling strength between them. This strength varies according to a STDP-type rule.

Research on the Kuramoto model with synaptic plasticity began in the early 2000s. In 2002, Seliger et al. [95] studied the Kuramoto model with slowly varying pair-wise coupling. The pair-wise coupling between oscillators varied according to a sinusoidal function of the phase difference of the oscillators and introduced learning rate and learning amplification parameters [95]. The coupling strengthened for small phase differences and weakened for large phase differences [95]. For a small learning rate, where the time-scale of the changes in the coupling strength was less than that of the change in the phases, the system formed synchronised clusters that were stable under external noise [95].

Maistrenko et al. [94] were the first to develop an adaptive Kuramoto model with coupling that varied according to the STDP-rule in Equation 1.1. They set the potentiation time constant to be less than the depression time constant [94]. They justified this choice of parameters using observations by [46, 98–100]. They introduced a learning rate and maximum learning parameter. Increasing the maximum learning parameter revealed three synchronisation regions. In the first region, which had the lowest maximum learning, there was no synchronisation. In the second region, where the maximum learning was intermediate, multistability was observed, in which partially synchronised states coexist with complete frequency-synchronised states. As a result, the system exhibits hysteresis: the transition to synchronisation occurs at a higher coupling strength than the transition back to the desynchronised state [94]. Importantly, these dynamics were only observed when the potentiation and depression constants were not equal. Fur-

ther, when the learning rate was increased, the multi-stability behaviour disappeared at a critical learning rate. At this rate, the model reverted to the behaviour of the classic Kuramoto model [94].

Following in the footsteps of Seleger et al. [95]., Ren et al. [101] also developed a Kuramoto model with coupling that adapts according to a sinusoidal function [101]. In this model, coupling strengthened for oscillators with large phase differences and weakened for oscillators with smaller phase differences, which is opposite the rule used by Seleger et al. [95]. The rule used by Ren et al. [101] had hard lower and upper bounds, producing dynamics with steady states that approached one of the two bounds, making the model suitable for storing binary information. They further observed that the model’s dynamics were retained for a network with local coupling and for larger systems [101].

Cumin and Unsworth [93] investigated the behaviour of the Kuramoto model with sinusoidally varying coupling, different coupling maps, and time-varying frequency. They found that increasing the number of connections for a constant positive coupling strength increased synchronisation, while increasing the number of connections with a constant negative coupling decreased synchronisation [93]. They argued that the positive coupling mapped to excitatory synapses while the negative coupling corresponded to inhibitory synapses [93]. For a slow sinusoidal time-varying coupling strength with constant natural frequencies and phases, slow quasi-synchronisation, characterised by spontaneous large drops in synchrony with a low drop frequency, was observed [93]. Furthermore, increasing the modulation frequency of the coupling strength led to faster quasi-synchronisation with more frequent drops in synchrony. In addition, when the frequency and phase of the time-varying coupling strength were randomized, the system exhibited non-synchronous behaviour [93].

Cumin and Unsworth [93] also found that time-varying natural frequencies produced stronger synchronisation effects than time-varying coupling strengths, with slow variations leading to quasi-synchrony and faster variations increasing the frequency of synchrony drops. They showed that combining time-varying natural frequencies and coupling strengths yielded quasi-synchronisation dependent on phase consistency, and that all-to-all connectivity was sufficient to enforce quasi-synchrony even for small networks [93]. They further demonstrated that increasing the number of oscillators strengthened quasi-synchrony into full synchrony, though with longer convergence times, and that combined effects exhibited superposition [93].

In 2009, Niyogi and English [102] studied the influence of the learning rate on the

emergent properties of the adaptive Kuramoto model used by Seleger et al. They found that for a fixed maximum coupling, increasing the learning rate past a critical rate produced two synchronous clusters instead of one [102]. Further, below the critical coupling of the non-adaptive Kuramoto model, the system did not attain global synchronisation for low learning rates [102].

In 2014, Timms and English [103] extended the model by Niyogi and English to consider the effects of time delay. They analytically and numerically investigated the dynamics of the model where the time delay represented the finite speed of signal propagation among oscillators in the network [103]. Their time delay was based on the spatial distance between connected oscillators and a constant signal speed [103]. They investigated synchronisation in 1D and 2D models. In the 1D model, they identified two-cluster integer and half-integer modes and derived synchronisation frequencies that were confirmed computationally [103]. The 2D model revealed that increasing the time delay altered system dynamics. In the static coupling 2D network, a fixed synchronisation frequency was observed. Under the dynamic coupling model, spatiotemporal coherent structures emerged with a single synchronisation, and the coupling network adapted to minimize the time delay [103].

Reviving the approach proposed by Maistrenko et al. [94], Ansariara et al. [104] investigated the effect of frustration on the Kuramoto model with STDP. Unlike Mainstrenko, their model employed identical potentiation and depression time constants [104]. They introduced further generalizations by introducing a local coupling between the oscillators of the model [104]. Their investigation was limited to small frustrations. In this frustration region, they observed phase-locked synchronisation with a pacemaker [104]. They also demonstrated that frustration introduces phase-linked hysteresis, where the phase differences and adjacency map of the systems can be recovered after a change in the frequency of one of the oscillators. This demonstrates the model’s capacity for retaining memory [104].

In 2022, Botha et al. [69] extended the work by Ansariara et al. [104] to investigate the emergence of chimera states. They observed chimera states for frustrations that were just less than $\frac{\pi}{2}$ and just greater than $3\frac{\pi}{2}$ [69]. This chimera state was non-stationary, with the synchronized oscillators changing over time [69]. This work showed that the frustration creates regions of in-phase synchronisation, phase-locked synchronisation, no synchronisation, and chimera states [69].

Collectively, this literature demonstrates that ensembles that synchronise via Kuramoto-type models with their connection strengths varying via STDP and STDP-rules exhibit

signs of memory, such as phase-locked states that are robust to noise after learning [95] and hysteresis [69, 104]. They also exhibit behaviour observed in actual neuronal ensembles, such as the emergence of pacemakers [104], formation of synchronizing cluster [95], and chimera states [69]. Hence, such models are promising models of memory encoding neuronal ensembles that can be synchronized to model mental synthesis.

1.2.4 Models of Multiple Neuronal Ensembles

This subsection introduces literature on the mathematical modelling of multiple neuronal ensembles. It seeks to review how existing models of multiple ensembles are built and gain clues into how a model of the synchronisation of memory encoding neuronal ensembles can be developed.

Present models of multiple neuronal ensembles assume a network of subnetworks structure, where individual ensembles are subnetworks with independent coupling rules and coupling between the subnetworks follow a different structure to form one larger network [105–107]. In addition, they use Hodgkin-Huxley equations to model the membrane potentials of neurons and the strength of synapses between neurons [105–107].

In 2022, Andreev et al. [105] developed a mathematical model of "network of networks" with a small input subnetwork that transmitted signals to four larger subnetworks with inter-subnetwork couplings. Their aim was to model the processes that occur in the brain during the processing of similar input information [105]. The input subnetwork had 5 neurons and the four larger subnetworks had 50 neurons each. The five neurons in the input subnetwork were all-to-all coupled. Each of the five neurons were randomly connected to the neurons of the subnetworks with a probability of 0.3 [105]. The larger subnetworks had a Watts–Strogatz small-world network topology [105], where each neuron was connected to neurons to its left and right and then these connections are reconnected with a certain probability avoiding heavy edges and self-loops [108]. Random bidirectional connections between neurons of the subnetworks were implemented randomly with a 0.3 probability of coupling [105]. Transmitting a signal via the input network showed an oscillation in the synchronisation indices of the larger subnetworks. The authors suggest that this oscillation represents the redistribution of processing loads between neurons of the different subnetworks [105]. They also found that maintaining this type of synchronisation required the balancing of intra and inter-subnetwork connections [105].

In 2023, Huang et al. [109] studied the synchronisation transitions of two connected neuronal ensembles with different sizes. One of the ensembles had five neurons with a connection probability of 0.3. The other ensemble had 50 neurons that were connected via the Watts–Strogatz small-world network [109]. Between the subnetworks, unidirectional connections from the smaller subnetwork to the larger subnetwork were made with a connection probability of 0.4 [109]. The neurons and synapses in this system were modelled using the Hodgkin–Huxley model [109]. Introducing time delays to this model produced multiple synchronisations in the network. Increasing coupling strength promoted synchronisation in the network when time delay was an integer multiple of the firing period of a single neuron [109]. When the subnetworks were internally well-synchronized, increasing time delay in both subnetworks induced multiple synchronisation transitions. Further, the synchronisation state of the smaller subnetwork influenced that of the larger subnetwork. Finally, increasing the time delay between the two subnetworks caused the synchronization factor of the modular network to vary periodically but had no effect on the synchronisation of the receiving network.

In 2025, Huang et al. [107] studied the emergence of chimera states in models of multiple neuronal ensembles. They constructed two subnetworks of 100 neurons that have the Watts–Strogatz small-world topology [107]. Unidirectional connections between neurons in the two subnetworks were made with a probability of 0.04 [107]. The Hodgkin–Huxley model was used to model the membrane potentials of the neurons and the strengths of the synapses between them [107]. Huang et al. used a technique called dynamical learning of synchronisation to alter the electromagnetic induction of the Hodgkin–Huxley model and study its effect on chimera states [107]. They observed that when the controlling sub-network has a chimera state, increasing the number of synchronized neurons stabilizes the synchronisation of the controlled sub-network. They found that these effects were robust against Gaussian noise [107].

The above studies provide a brief but instructive window into how to model the interaction of two or more neuronal ensembles. First, independent ensembles with their own network topologies must be created. Next, these networks can be connected through connections between neurons of the individual ensembles [105, 107, 109]. These connections can be unidirectional, where one ensemble controls another [107, 109], or bidirectional, where the ensembles mutually influence each other [105]. However, these studies solely rely on the Hodgkin–Huxley model for modelling interactions between neurons, and the work has not directly focused on investigating a particular brain function that requires the integration of two or more neuronal ensembles.

1.3 Problem Statement and Methodology

This section introduces the problem addressed in this dissertation and outlines the methodological approach used to investigate it. Section 1.3.1 presents the problem statement, highlighting the lack of a mathematical model of the neuronal dynamics underlying mental synthesis. The objectives of the study are then specified to clarify the scope of the proposed modelling framework. Following this, Section 1.3.2 describes the numerical procedures used to construct, validate, and analyse the developed models. These components establish the foundation for the computational work presented in the subsequent chapters.

1.3.1 Problem Statement

Mental synthesis is the voluntary combination of mental images to form novel mental imagery under the control of the LPFC [1, 2, 17]. The formed imagery is not restricted to visual images; rather, it may incorporate auditory, gustatory, olfactory, or other sensory modalities [1]. A central hypothesis in the literature is that mental synthesis occurs when neuronal ensembles that encode the memories of the synthesized imagery are synchronised [1, 2]. The memory storing ensembles are made up of neurons that are distributed across cortical regions [28], with highly selective neurons located in the temporal lobe and less selective neurons residing in earlier visual areas [34].

Over the years, several mathematical models that combine STDP and phase synchronisation through Kuramoto-type oscillatory dynamics have been proposed. These models capture the dynamics of memory encoding neuronal ensembles. They have demonstrated the ability to store and retain memories, suggesting that they may serve as useful abstractions of memory-encoding neuronal ensembles [95, 101, 102, 104]. Recent work has also shown that multiple ensemble models may be coupled to study how the brain processes information across networks of ensembles [105].

Despite these advances, a mathematical model of mental synthesis that combines memory encoding and the synchronisation of distributed ensembles has not been reported. Such a model must capture the formation of synaptic maps through synaptic plasticity while also enabling synchronisation between ensembles. Importantly, this synchronisation must occur without disrupting the phase-based structures and synaptic connections that define the individual elements being combined.

Most existing STDP-based models do not address this problem because they rely on simplified network structures. Many assume either all-to-all or cyclic, nearest-neighbour connectivity [93–95, 101–103]. Although analytically convenient, these architectures do not reflect the distributed connectivity patterns observed in memory formation and recall. In addition, these models are typically examined in isolation, and their collective behaviour under inter-ensemble coupling has not been explored.

In this dissertation, these gaps are addressed by developing a model of mental synthesis. The model is based on neuronal ensembles with inter-neuron connections changing via STDP. It extends the plasticity framework introduced by Maistrenko et al. [94] and later refined by Ansariara et al. [104] and Botha et al. [69]. To improve the biophysical plausibility of the framework, we replace cyclic coupling with random coupling to realize distributed and sparse connectivity. Finally, to model mental synthesis, we introduce bidirectional random connections between two such ensembles and assess whether they can synchronise while retaining elements of their phase and synaptic connectivity structures.

The significance of this work lies in its contribution to improving the realism of mathematical models of memory-encoding neuronal ensembles that are based on STDP. The study introduces distributed connectivity and inter-ensemble coupling to develop a framework for computationally testing the hypothesis of the neuronal mechanism underlying mental synthesis proposed by Vyshedsky [1]. The developed approach provides a basis for future theoretical and empirical investigations into how memories are integrated in the brain to generate novel mental content.

This study has the following objectives:

1. To extend the Kuramoto–Sakaguchi STDP-based neuronal ensemble model proposed by Ansariara et al. [104] by introducing a distributed network topology in which each neuron has a fixed node degree and is equally likely to connect to any other neuron in the ensemble.
2. To investigate how random connectivity influences synchronisation dynamics and memory encoding.
3. To develop a model of mental synthesis by connecting two distributed ensembles through random, bidirectional inter-ensemble couplings.
4. To identify parameter regimes that allow the two ensembles to synchronise without the loss of previously encoded memories.

1.3.2 Methodology

The methodology of this study is based on numerical simulations. The models described above were constructed and simulated numerically to investigate their dynamical behaviour.

Because the implementations of the above models require advanced programming techniques and are quite complex, it is essential to verify the accuracy of the implementations at each stage of development. In this study, we did so systematically by validating our simulation framework against known results. We thus first reproduced classical results for the Kuramoto model, focusing on experiments of the onset of synchronisation, and then extended these tests to the Kuramoto–Sakaguchi model. This step ensured that our numerical methods were consistent with the theoretically expected behaviours.

We then reproduced some results from Maistrenko et al. [94], focusing on reproducing the emergence of bistability for an asymmetric model of STDP. We also reproduce findings from Ansariara et al. [104] and Botha et al. [69], where ensemble dynamics were studied under cyclic coupling. These replications provided a solid foundation for the implementation and evaluation of the new model variants that were subsequently developed.

Following this validation stage, we implemented a distributed topology by replacing cyclic coupling with random connectivity. We conducted numerical experiments to examine how this change affects synchronisation behaviour and the properties of the steady states of the model.

Finally, we developed the coupled-ensemble model intended to capture mental synthesis. In this model, two ensembles with distributed connectivity are linked through a set of randomly assigned, bidirectional inter-ensemble connections. We explored the parameter landscape to identify conditions that support synchronisation between the ensembles while preserving the phase and synaptic structures established within each ensemble.

Chapter 2

Mathematical Details of the Model

One of the main objectives of this dissertation is to develop a mathematical model of mental synthesis. This model is developed within a phase reduction framework, which requires the combination of several component models. This chapter describes these models in their simplest form.

Section 2.1 describes a model of STDP that is compatible with the phase reduction framework. Section 2.2 describes the Kuramoto model and its fundamental dynamics. The Kuramoto–Sakaguchi model, which includes the frustration (or phase-lag) parameter, is described in Section 2.3. In Section 2.4, this model is extended to include adaptive coupling through STDP. Section 2.5 further generalises the model, relaxing the assumption of all-to-all coupling and investigating the effects of cyclic and random coupling. Finally, Section 2.6 introduces the model of coupled ensembles and provides the final equations governing the connectivity and dynamics of two coupled ensembles.

2.1 Phase Reduction Formulation of STDP

Section 1.2.1 showed that STDP can be modelled using Equation (1.1). Here, we reformulate the model in terms of oscillator phases to make it suitable for a phase reduction framework. First, the synaptic weight between neurons i and j , where i and j are the indices of the pre- and postsynaptic neurons, respectively, is defined as C_{ij} .

The evolution of this weight is defined as

$$\dot{C}_{ij} = \begin{cases} \epsilon(A - C_{ij})e^{-\frac{\theta_j - \theta_i}{\tau_p}}, & 0 \leq \theta_j - \theta_i \leq \pi, \\ -\epsilon C_{ij}e^{-\frac{\theta_j - \theta_i}{\tau_d}}, & -\pi \leq \theta_j - \theta_i < 0, \end{cases} \quad (2.1)$$

where $\dot{C}_{ij}$ is the instantaneous change in the synaptic weight, ϵ is the learning rate, A is the maximum synaptic weight, τ_p and τ_d are the potentiation and depression time constants, and θ_i and θ_j are the phases of the pre- and postsynaptic neurons, respectively.

This definition enables the use of the STDP rule in a phase reduction framework and eliminates the need to simulate spike times. Instead, changes in the synaptic weights are determined by phase differences between oscillators, where θ_i represents the phase of a tonically spiking neuron with frequency $\dot{\theta}_i$. In this definition, potentiation occurs when $0 \leq \theta_j - \theta_i \leq \pi$ and depression occurs when $-\pi \leq \theta_j - \theta_i < 0$.

The maximum potentiation is defined to avoid infinite potentiation, which may occur when oscillators maintain a positive phase difference. This change constrains the synaptic weights to a range of $[0, A]$. The learning rate ϵ determines the rate at which synaptic weights are updated. The potentiation and depression time constants τ_p and τ_d , respectively, determine the temporal windows over which potentiation and depression occur.

The present model describes only the change in synaptic weights under the assumption of tonically spiking neurons. For a more complete description of an ensemble of tonically spiking neurons, the evolution of the phases θ_i must be defined. This is achieved using the Kuramoto model, which is described next.

2.2 Kuramoto Model

This section reintroduces the Kuramoto model discussed in Section 1.2.2. While Section 1.2.2 focused on its historical development, this section examines its mathematical details and the transition from disorder to synchronisation.

In the standard Kuramoto model introduced in Equation (1.2), the natural frequencies ω_i are drawn from a random distribution $g(\omega)$ [65], introducing disorder into the system, as different oscillators evolve at different intrinsic frequencies. The model describes how this disorder is overcome by the coupling K .

The coupling K is assumed to be weak relative to the internal stability of the oscillators' limit cycles [64]. Under this assumption, interactions between oscillators affect only their phases and do not perturb the amplitudes and shapes of their limit cycles. Sufficiently strong coupling would alter the limit cycle amplitudes and shapes, which would require a more general description that involves phase and amplitude dynamics. Therefore, the weak coupling assumption justifies the phase reduction approach used in the Kuramoto model [78]. In the model, the scaling factor $1/N$ is introduced to ensure that the system has weak coupling regardless of the number of oscillators in the system.

The weak coupling acts as a restorative force between oscillators with different phases. When oscillator i leads oscillator j ($\theta_i > \theta_j$), the term $\sin(\theta_j - \theta_i)$ is negative, reducing the frequency of oscillator i . Conversely, if i lags oscillator j ($\theta_i < \theta_j$) the term $\sin(\theta_j - \theta_i)$ is positive, increasing the frequency of oscillator i . This adaptation acts to align the phases of the oscillators in the system.

To quantify the competing effects of frequency disorder and order induced by the weak coupling K , Kuramoto introduced the complex order parameter $Z(t)$:

$$Z(t) = r \exp(i\psi) = \frac{1}{N} \sum_{j=1}^N \exp(i\theta_j), \quad (2.2)$$

where $r(t) \in [0, 1]$ is the global coherence and ψ is the average phase of the oscillators. By multiplying both sides of Equation 2.2 by $\exp(-i\theta_i)$ and considering only the imaginary part, Equation 2.2 can be written in mean-field form:

$$\dot{\theta}_i = \omega_i + Kr \sin(\psi - \theta_i) \quad (2.3)$$

This representation shows that the oscillators are coupled to the average phase with a strength proportional to the global coherence r .

Assuming a natural frequency distribution $g(\omega)$ that is unimodal and symmetric about a mean frequency ω_0 makes it possible to transform the system into a rotating frame of reference with $\theta \rightarrow \theta - \omega_0 t$ [80]. This transformation sets the mean frequency to zero. In this way, the collective phase ψ becomes a constant, simplifying the analysis of the steady state [64]. Further, using this assumption and considering a system with $N \rightarrow \infty$, produces the critical coupling given in Equation (1.4).

The transition at K_C is characterised by the emergence of two populations of oscillators. As K approaches K_C , oscillators with natural frequencies satisfying $|\omega_i| \leq Kr$

are synchronised and contribute to the continuous growth of r [64]. In contrast, oscillators with $|\omega_i| > Kr$ are not entrained because the coupling strength is insufficient to overcome their intrinsic dynamics [64]. As the coupling strength K increases beyond K_C , an increasing fraction of the oscillators becomes entrained [64]. This progressive entrainment leads to a continuous growth of r from 0 towards 1.

2.3 Kuramoto–Sakaguchi Model

The previous section discussed the mathematical details of the basic Kuramoto model. In this section, we discuss the mathematical details of the Kuramoto–Sakaguchi model. As discussed in Section 1.2.2, this model introduces phase lag or frustration to the Kuramoto model [82] and describes systems where the interaction between oscillators is not instantaneous or perfectly attractive [110].

The phase lag α introduced in the Kuramoto–Sakaguchi model the dynamics of the model Kuramoto model. While the standard Kuramoto model always evolves toward a state of maximum alignment, the phase lag breaks this symmetry. This symmetry breaking results in a system where the oscillators do not necessarily evolve to maximum alignment [82].

Using the complex order parameter defined in Equation 2.2, the Kuramoto–Sakaguchi model can be written in mean-field form as

$$\dot{\theta}_i = \omega_i + Kr \sin(\psi - \theta_i - \alpha). \quad (2.4)$$

This representation shows that instead of the oscillators being entrained to the average phase ψ , they are entrained to the shifted phase $\psi - \alpha$. Because of the shift caused by the phase lag α , the oscillators maintain a phase offset with the average phase even when synchronised.

The introduction of α alters the threshold for the transition to synchronisation. For oscillators with natural frequencies following the unimodal distribution $g(\omega)$, the critical coupling is as given in Equation (1.6). As $|\alpha|$ increases toward $\pi/2$, $\cos(\alpha)$ decreases and K_C increases. This shows that increasing α increases the quantity of coupling required to bring the system into synchrony [82]. As $|\alpha| \rightarrow \pi/2$, K_C approaches infinity, indicating that the system cannot synchronise for $\alpha = \pm\pi/2$ [82]. When $|\alpha| > \pi/2$, K_C is negative, indicating that the system is repulsive. That is, the oscillators push away from one another, preventing full synchronisation regardless of the coupling strength.

When there is no disorder in the natural frequencies of the oscillators, i.e., $\omega_i = \Omega$ for all i , the behaviour of the system is entirely governed by the value of α . If $|\alpha| < \pi/2$, the coupling between the oscillators remains attractive and the system evolves to $r = 1$ for any $K > 0$. However, if $|\alpha| > \pi/2$, the coupling becomes repulsive and the system does not form a coherent cluster regardless of the value of K .

2.4 Kuramoto–Sakaguchi Model with STDP

The above two sections discussed the mathematical details of the Kuramoto and Kuramoto–Sakaguchi models. This section generalises the Kuramoto–Sakaguchi model by moving away from a constant coupling K to adaptive coupling. This adaptive coupling is governed by the STDP rule introduced in Section 2.1. This generalisation is introduced to mimic neuronal ensembles that have synapses with weights that vary via STDP.

The generalisation introduced in this section takes the form

$$\dot{\theta}_i = \omega_i + \sum_{j=1}^N \frac{C_{ij}}{N} \sin(\theta_j - \theta_i - \alpha), \quad (2.5)$$

where $C_{i,j}$ adapts via Equation (2.1).

In Equation 2.1, the learning rate is set such that $\epsilon \ll 1$, introducing a time scale separation between the evolution of the coupling weights and the evolution of the phases [95]. Under this timescale difference, the phases evolve much faster than the coupling weights, settling to a phase-locked state that the network structure can learn. This mechanism is analogous to synaptic plasticity in biological neural systems, where learning is driven by correlations in spike timing. In the phase-reduced description, these temporal correlations are represented by phase relationships, suggesting that phase-based learning is analogous to STDP in real neuronal networks.

The system with STDP has more dimensions than the systems described in the previous two sections. Instead of a system with N equations that govern the evolution of the oscillator phases, the new system has N^2 equations, with N equations describing the evolution of the phases and $N(N - 1)$ equations describing the evolution of the coupling between the oscillators. The added dimensions transform the problem from a simple phase-coherence problem into a network problem, where the STDP rule strengthens connections between oscillators that are already nearly synchronised and weakens connections between oscillators that are not.

The introduction of adaptive coupling changes the transition to synchrony. Instead of tuning the parameter K , the parameter A can be tuned for fixed ϵ , τ_p , and τ_d to determine the onset of synchrony. A careful consideration of the onset of synchrony for this system has not been published. However, a study based on numerical simulations has shown that the transition depends on the symmetry of the STDP rule. For a symmetric rule, where $\tau_d = \tau_p$, a single transition to synchrony is observed with increasing A . However, when the rule is asymmetric, multistability is observed, where for certain values of A , stable desynchronised states coexist with the fully synchronised state [94].

The introduction of the adaptive coupling makes the network of oscillators a directed graph because $C_{ij} \neq C_{ji}$. The directed network further breaks the symmetry of the standard Kuramoto model.

2.5 Modification of the Coupling Structure

The above sections presented the governing equations of models where the oscillators are all-to-all coupled. That is, each oscillator was coupled to all the other oscillators in the system. This coupling architecture is not realistic for modelling most physical systems. This section presents the mathematical details of two network structures that are more plausible for modelling physical systems, namely, cyclic and random coupling.

The cyclic network has been used to model neuronal ensembles with synaptic weights that adapt via STDP [69, 104]. For this architecture, each oscillator has connections only with its nearest neighbours. This connectivity is implemented as follows:

$$G_{ij} = \begin{cases} 1, & 1 \leq |i - j|_{\text{mod}, N} \leq \frac{n}{2} \\ 0, & \text{otherwise,} \end{cases} \quad (2.6)$$

where G_{ij} is the edge from oscillator i to oscillator j . When $G_{ij} = 1$, an edge from node i to node j exists. If $G_{ij} = 0$, an edge from oscillator i to oscillator j does not exist. The index distance $|i - j|_{\text{mod}, N}$ is defined modulo N to ensure cyclic boundary conditions. The lower bound connectivity 1 is defined to exclude self coupling, since $|i - j|_{\text{mod}, N} = 0$ if $i = j$. The upper bound $n/2$ ensures that each oscillator is connected to its $n/2$ nearest neighbours on either side. Consequently, each oscillator has n connections. Hence, n is referred to as the node degree. Here, n is even to ensure the coupling matrix symmetric.

For a system with $N = 12$ oscillators and $n = 6$, this implementation produces the

adjacency matrix

$$G = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}. \quad (2.7)$$

This matrix creates a connection topology where each oscillator is cyclically connected to its nearest neighbours. However, it does not capture the presence of long-distance connections between oscillators. This limitation can be addressed by introducing a less structured network based on random connectivity.

Here, a random network in which each oscillator has a fixed node degrees n is considered. In this network, each oscillator is equally likely to be connected to any other oscillator in the network while preserving symmetry. The network is defined such that if $G_{ij} = 1$, then $G_{ji} = 1$, and

$$G_{ij} = \begin{cases} 1, & i \neq j \text{ and } j \in \mathcal{N}_i, \\ 0, & \text{otherwise,} \end{cases} \quad (2.8)$$

where $\mathcal{N}_i \subset \{1, \dots, N\} \setminus \{i\}$ is a randomly chosen set of cardinality $|\mathcal{N}_i| = n$ subject to the symmetry constraint

$$j \in \mathcal{N}_i \iff i \in \mathcal{N}_j.$$

An example adjacency matrix produced by this implementation is given below:

$$G = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \end{pmatrix} \quad (2.9)$$

This adjacency matrix allows for both short- and long-range connections, while maintaining a fixed node degree. It produces a topology that is less regular and therefore more biologically realistic.

In the model presented in Section 2.4, the presented adjacency matrices constrain the STDP. That is, $C_{ij} = 0$ if $G_{ij} = 0$ and evolves according to Equation (2.1) if $G_{ij} = 1$. The following examples illustrate how the adjacency matrix G constrains the adaptive coupling matrix C for both cyclic and random connectivity. The Examples are given for $N = 8$ and $n = 4$

$$G^{\text{cyc}} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \quad (2.10)$$

$$C^{\text{cyc}} = \begin{pmatrix} 0 & C_{12} & C_{13} & 0 & 0 & 0 & C_{17} & C_{18} \\ C_{21} & 0 & C_{23} & C_{24} & 0 & 0 & 0 & C_{28} \\ C_{31} & C_{32} & 0 & C_{34} & C_{35} & 0 & 0 & 0 \\ 0 & C_{42} & C_{43} & 0 & C_{45} & C_{46} & 0 & 0 \\ 0 & 0 & C_{53} & C_{54} & 0 & C_{56} & C_{57} & 0 \\ 0 & 0 & 0 & C_{64} & C_{65} & 0 & C_{67} & C_{68} \\ C_{71} & 0 & 0 & 0 & C_{75} & C_{76} & 0 & C_{78} \\ C_{81} & C_{82} & 0 & 0 & 0 & C_{86} & C_{87} & 0 \end{pmatrix} \quad (2.11)$$

$$G^{\text{rand}} = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \quad (2.12)$$

$$C^{\text{rand}} = \begin{pmatrix} 0 & C_{12} & 0 & C_{14} & 0 & C_{16} & 0 & C_{18} \\ C_{21} & 0 & C_{23} & 0 & C_{25} & 0 & 0 & C_{28} \\ 0 & C_{32} & 0 & C_{34} & C_{35} & 0 & C_{37} & 0 \\ C_{41} & 0 & C_{43} & 0 & 0 & C_{46} & C_{47} & 0 \\ 0 & C_{52} & C_{53} & 0 & 0 & C_{56} & 0 & C_{58} \\ C_{61} & 0 & 0 & C_{64} & C_{65} & 0 & C_{67} & 0 \\ 0 & 0 & C_{73} & C_{74} & 0 & C_{76} & 0 & C_{78} \\ C_{81} & C_{82} & 0 & 0 & C_{85} & 0 & C_{87} & 0 \end{pmatrix} \quad (2.13)$$

In both cases, the non-zero structure of C matches that of G , so that synaptic weights evolve only along existing connections.

The constraint introduced by the discussed connectivity matrices reduces the dimensionality of the Kuramoto–Sakaguchi model with STDP. Instead of N^2 equations, the system now has $N(n+1)$ equations. Further, the phase dynamics are given by

$$\dot{\theta}_i = \omega_i + \sum_{j=1}^N \frac{C_{ij}}{n} \sin(\theta_j - \theta_i - \alpha), \quad (2.14)$$

where the coupling strengths are normalised by the node degree to ensure that the system does not diverge for large systems with large node degrees.

2.6 Coupled Neuronal Ensembles

The above discussion considers only a single neuronal ensemble. Mental synthesis, which is modelled in this study, requires the integration of multiple memories. Since memories are assumed to be encoded in neuronal ensembles, a model of mental synthesis requires the modelling of two or more ensembles.

Therefore, this section introduces the mathematical details of two coupled ensembles with random connectivity. This connectivity was selected because it reflects the distributed and non-local connectivity observed in neuronal ensembles involved in brain function.

To represent the system, two neuronal ensembles, each with N neurons, are considered. The network with both ensembles has neurons with the indices $i \in \{1, \dots, 2N\}$, where the first and second ensembles correspond to the index sets $\mathcal{E}_1 = \{1, \dots, N\}$ and $\mathcal{E}_2 = \{N+1, \dots, 2N\}$, respectively. The connectivity of the system is described by the adjacency matrix $\mathcal{G}$ of size $2N \times 2N$ that is partitioned as follows:

$$\mathcal{G} = \begin{pmatrix} G^{(1)} & G^{(1,2)} \\ G^{(2,1)} & G^{(2)} \end{pmatrix} \quad (2.15)$$

where $G^{(1)}$ and $G^{(2)}$ describe the intra-ensemble connectivity of the individual ensembles and $G^{(1,2)}$ and $G^{(2,1)}$ represent the inter-ensemble connections. Since we assume bidirectional connectivity, $\mathcal{G}_{ij} = \mathcal{G}_{ji}$, implying $G^{(2,1)} = G^{(1,2)^T}$.

For each of the ensembles, the intra-ensemble adjacency matrix is defined as

$$G_{ij}^\sigma = \begin{cases} 1, & i \neq j \text{ and } j \in \mathcal{N}_i^\sigma, \\ 0, & \text{otherwise,} \end{cases} \quad (2.16)$$

where $\sigma \in \{1, 2\}$ is the ensemble index, $i, j \in \mathcal{E}_\sigma$, and $\mathcal{N}_i^\sigma \subset \mathcal{E}_\sigma \setminus \{i\}$ is a randomly chosen set of cardinality $|\mathcal{N}_i^\sigma| = n$ satisfying the symmetry condition

$$j \in \mathcal{N}_i^\sigma \iff i \in \mathcal{N}_j^\sigma,$$

ensuring that each ensemble has symmetric connectivity and a neuron node degree of n .

The inter-ensemble connections are defined as

$$G_{i,p}^{(1,2)} = \begin{cases} 1, & p \in \mathcal{M}_i, \\ 0, & \text{otherwise,} \end{cases} \quad (2.17)$$

where $i \in \mathcal{E}_1$, $p \in \mathcal{E}_2$, and $\mathcal{M}_i \subset \mathcal{E}_2$ is a randomly selected set of neurons in the second ensemble with cardinality $|\mathcal{M}_i| = m$, making the parameter m the inter-ensemble node degree. To ensure bidirectionality in the connections between the two ensembles, $G_{pi}^{(1,2)} = G_{ip}^{(2,1)}$.

The adjacency matrix constrains the possible coupling matrix $\mathcal{C}$ of the two-ensemble system. That is, $\mathcal{C}_{ij} = 0$ if $\mathcal{G}_{ij} = 0$. An example of this constraint is provided below for $N = 5$, $n = 2$, and $m = 2$, where $\mathcal{G}$ and $\mathcal{C}$ have the shape 10×10 :

$$\mathcal{G} = \left(\begin{array}{ccccc|ccccc} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{array} \right) \quad (2.18)$$

$$\mathcal{C} = \left(\begin{array}{ccccc|ccccc} 0 & 0 & 0 & C_{1,4} & C_{1,5} & 0 & 0 & C_{1,8} & 0 & C_{1,10} \\ 0 & 0 & C_{2,3} & 0 & C_{2,5} & 0 & 0 & 0 & C_{2,9} & C_{2,10} \\ 0 & C_{3,2} & 0 & C_{3,4} & 0 & C_{3,6} & 0 & 0 & C_{3,9} & 0 \\ C_{4,1} & 0 & C_{4,3} & 0 & 0 & 0 & C_{4,7} & C_{4,8} & 0 & 0 \\ C_{5,1} & C_{5,2} & 0 & 0 & 0 & C_{5,6} & C_{5,7} & 0 & 0 & 0 \\ \hline 0 & 0 & C_{6,3} & 0 & C_{6,5} & 0 & 0 & C_{6,8} & 0 & C_{6,10} \\ 0 & 0 & 0 & C_{7,4} & C_{7,5} & 0 & 0 & 0 & C_{7,9} & C_{7,10} \\ C_{8,1} & 0 & 0 & C_{8,4} & 0 & C_{8,6} & 0 & 0 & C_{8,9} & 0 \\ 0 & C_{9,2} & C_{9,3} & 0 & 0 & 0 & C_{9,7} & C_{9,8} & 0 & 0 \\ C_{10,1} & C_{10,2} & 0 & 0 & 0 & C_{10,6} & C_{10,7} & 0 & 0 & 0 \end{array} \right) \quad (2.19)$$

The block structure of the matrices highlights the intra-ensemble connectivity along the diagonal blocks and inter-ensemble connections in the off-diagonal blocks. An example of this connectivity is given in Figure 2.1.

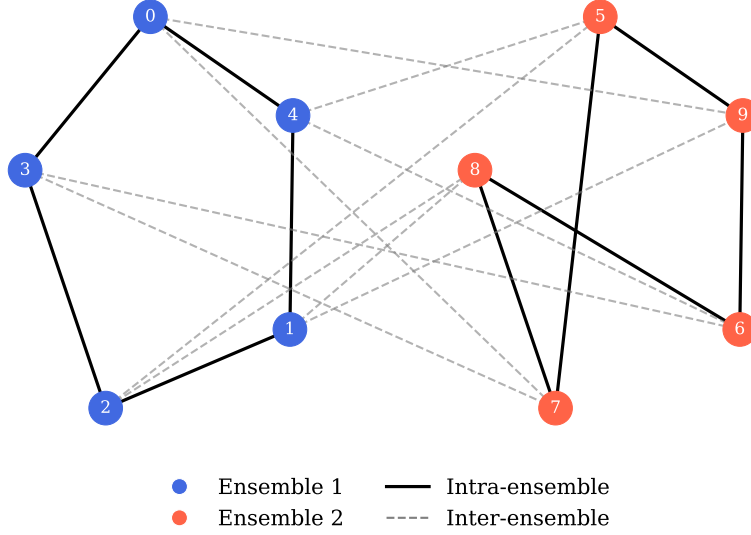


Figure 2.1: Example of two networks with intra-ensemble and inter-ensemble coupling. The nodes of ensemble 1 are indicated in blue and those of ensemble 2 are indicated in orange. Intra-ensemble couplings are indicated by solid black lines and inter-ensemble couplings by grey dashed lines.

For the STDP rule, this connectivity matrix introduces $2Nm$ new pairwise couplings. Further, the instantaneous frequencies of the individual neurons become

$$\dot{\theta}_i = \omega_i + \begin{cases} \sum_{j=1}^N \frac{C_{ij}}{n} \sin(\theta_j - \theta_i - \alpha) + \sum_{p=N+1}^{2N} \frac{C_{ip}}{m} \sin(\theta_p - \theta_i - \alpha), & i \in \mathcal{E}_1 \\ \sum_{j=N+1}^{2N} \frac{C_{ij}}{n} \sin(\theta_j - \theta_i - \alpha) + \sum_{p=1}^N \frac{C_{ip}}{m} \sin(\theta_p - \theta_i - \alpha). & i \in \mathcal{E}_2 \end{cases} \quad (2.20)$$

In the equation, the first summation describes intra-ensemble coupling, while the second describes inter-ensemble interactions.

The dimensionality of this system is $2N(n + m + 1)$, and it describes the phase dynamics of the oscillators in the ensembles under the influence of both intra- and inter-ensemble coupling.

This chapter has detailed the mathematical models that will be used in this study. It introduced the STDP rule adapted for compatibility with the phase reduction framework employed in this study. The details of the Kuramoto model were discussed, and it was generalised into the Kuramoto–Sakaguchi model with STDP under cyclic and random coupling. Finally, a model that describes the connectivity and dynamics of neurons in a network of two ensembles was introduced and detailed. The next chapter describes how we developed scripts for simulating these models. It also details the simulation procedures that were used to investigate the dynamics of the models.

Chapter 3

Code Development and Preliminary Benchmarking Simulations

This chapter shows the development of the Python functions used to simulate Eqs. 2.5 and 2.20. To develop these functions, we first constructed a function that simulates the basic Kuramoto model and then systematically introduced additional parameters and complexity. As we introduced more parameters and complexity, we reproduced known results to validate our functions. These reproduced results are also used as references for the novel results presented in Chapters 4 and 5.

In Section 3.1, we develop a function for simulating the classical Kuramoto model presented in Section 2.2. We use this function to successfully produce finite-size versions of the curve that demonstrates the onset of synchrony in the Kuramoto model at critical coupling. We also show that a pure Python approach with nested loops introduces computational complexity that makes the wall time of our calculations impractical, particularly for large systems. We introduce just-in-time compilation via the Numba library to overcome this bottleneck.

Section 3.2 introduces the frustration term into the function developed in Section 3.1. This modification produces a function that simulates the Kuramoto–Sakaguchi model. We show that the resulting function can reproduce the shift in the critical coupling required to induce synchronisation observed for frustrations in the range $(-\pi/2, \pi/2)$.

Having established a working function for the Kuramoto and Kuramoto–Sakaguchi models, in Section 3.3, we introduce STDP coupling into the function. To test the validity of this implementation, we reproduce a result presented by Maistrenko et al.,

showing that the Kuramoto model with STDP coupling exhibits multistability when the potentiation and depression time constants are not equal [94].

Next, in Section 3.4, we vary the connectivity topology by introducing helper functions that implement cyclic coupling. This modification effectively reproduces the model presented by Ansariara et al. [104]. We use this section to reproduce findings reported by Ansariara et al. [104] and Botha et al. [69].

Section 3.5 presents the functions used to model the Kuramoto–Sakaguchi model with STDP and random coupling. These functions are used to simulate Equation 2.5 with random coupling in order to investigate the behaviour of ensembles composed of distributed neurons. Finally, to close the chapter, Section 3.6 presents the functions used to simulate Equation 2.20. These simulations are used to investigate the behaviour of a system of two connected ensembles with distributed neurons.

3.1 Developing a Python Function for Simulating the Kuramoto Model

This section describes the development of a function that will be used to simulate the Kuramoto model. We first transcribe the governing equations of the model into Python and validate the accuracy of our transcription by reproducing the theoretical onset of synchronisation curve. In the process of validating our transcription, we encounter a computational challenge. When we increase the system size, the computational cost of the direct implementation becomes impractical. To overcome this challenge, we introduced JIT compilation to accelerate our computations. We show that JIT enables the simulation of systems with larger sizes that would otherwise be impractical through the pure Python approach while faithfully reproducing the expected model dynamics.

3.1.1 Direct Simulation of the Kuramoto Model

The Kuramoto model, Equation (1.2), was transcribed as the following Python function:

```
def kuramoto_model(t, y, w, K, N):
    yd = np.zeros(N)
    for i in range(N):
```

```

for j in range(N):
    theta = y[j] - y[i]
    yd[i] += K * np.sin(theta)
yd[i] = w[i] + yd[i] / float(N)

return yd

```

The function has the arguments `t` for time, `y` and `w` to store the oscillator phases and natural frequencies, respectively, `K` for the value of the coupling strength, and `N` for the number of oscillators. The argument `t` is not used in the function but is included as the leading argument because it is crucial for the integration routine used in our experiments. `yd` is created as an array that stores the instantaneous frequencies of the oscillators and is returned by the function at each integration time step.

The function’s nested loop structure reflects the all-to-all coupling topology of the standard Kuramoto model. That is, the instantaneous frequency of each oscillator i is cumulatively influenced by each oscillator j . This cumulative influence is then normalized by the system size N before being added to the oscillator’s natural frequency ω_i . This implementation is a direct numerical translation of the model definition without approximation or reduction, making it ideal for validation against known results.

3.1.2 Numerical Integration Details and Order Parameter

For our integration, we used SciPy’s `solve_ivp` routine with an adaptive high-order integrator (DOP853). The initial phases were sampled from a uniform distribution with the range of $[-\pi, \pi]$, and the natural frequencies were drawn from a Gaussian distribution $\mathcal{N}(\mu, \sigma)$. We used the Kuramoto order parameter to quantify the degree of collective synchrony:

$$r(t) = \frac{1}{N} \left| \sum_{j=1}^N \exp(i\theta_j(t)) \right|, \quad (3.1)$$

where i is the imaginary number and $r(t) \approx 0$ indicates incoherence and $r(t) \rightarrow 1$ indicates strong coherence. We estimated the steady-state $\langle r \rangle$ by discarding an initial transient portion of the time series of r and averaging over the remaining time window. This estimation was performed for each simulation.

3.1.3 Onset of Synchrony Validation

To test our transcription of the Kuramoto model, we performed simulations that verify the critical coupling K_C introduced in Equation (1.4). For a Gaussian frequency distribution with standard deviation σ ,

$$g(0) = \frac{1}{\sqrt{2\pi}}, \Rightarrow K_c = \frac{2\sqrt{2\pi}\sigma}{\pi}. \quad (3.2)$$

For simplicity, we selected a $\sigma = 0.63$. This parameter choice ensured that $K_c \approx 1.005$. We then performed simulations for systems of various sizes, namely, $N = 20, 50, 100$, and 150 , where the absolute and relative error tolerances of the solver were both set to $1e-12$, the integration time was $T = 50$, and the sample time was $dt = 0.1$ using the following implementation:

```
def single_experiment(N, K, mu, sigma, T, dt):
    w0 = np.zeros(N, 'd')
    w0 = np.random.normal(mu, sigma, N)
    y0 = np.zeros(N, 'd')
    y0[0:N] = np.random.uniform(-np.pi, np.pi, N)
    abt = 1e-12 #setting tolerences
    ret = 1e-12 #setting tolerences
    t_eval = np.arange(0, T, dt)
    sol = solve_ivp(kuramoto_model, [0, T], y0, args=(w0, K, N),
                    method='DOP853', atol = abt, rtol = ret,
                    t_eval=t_eval)
    y = sol.y.T
    r_t = np.abs(np.sum(np.exp(1j * y), axis=1)/N)

    return(r_t)

def sweep_K_vs_r(N_values, K_values, mu=0, sigma=1,
                  T=50, dt=0.1, transient_fraction=0.5):

    results = {}

    for N in N_values:
```

```

    r_mean_list = []

    print(f"Running simulations for N = {N}")

    for K in K_values:
        r_t = single_experiment(N, K, mu, sigma, T, dt)

        # discard transient
        cut = int(len(r_t) * transient_fraction)
        r_ss = np.mean(r_t[cut:])

        r_mean_list.append(r_ss)

    results[N] = np.array(r_mean_list)

    return results

if __name__ == "__main__":

    N_values = [20, 50, 100, 150]
    K_values = np.linspace(0, 5, 25)

    results = sweep_K_vs_r(N_values, K_values)

```

where the function `single_experiment` performs a single simulation of the Kuramoto model for a system of size N . It generates the natural frequencies from a normal distribution with mean μ and standard deviation σ , and selects the initial oscillator phases from a uniform distribution on $[-\pi, \pi]$. The system is then integrated using `solve_ivp` with the DOP853 method, and the order parameter $r(t)$ is calculated at each sampling time. The function `sweep_K_vs_r` performs simulations for a range of coupling strengths K and system sizes N . For each simulation, the transient portion of the order parameter time series is discarded and the mean steady-state value $\langle r \rangle$ is calculated. These values are then stored for each system sizes N . The `sweep_K_vs_r` function is then used in the main execution block for the system sizes of $N = 20, 50, 100$, and, 150 and a range of coupling strengths $K \in [0, 5]$ sampled at 25 evenly spaced points. It is called to compute the steady-state order parameter for each combination of N and K .

We compared the computed theoretical K_c and the numerically observed transition.

This comparison is shown in Figure 3.1. The curves in Figure 3.1 show a transition occurring near the predicted value and an increasingly sharp transition with increasing N , consistent with the behaviour of $N \rightarrow \infty$. This agreement provides evidence that the baseline implementation accurately reproduces the expected dynamics of the Kuramoto model. However, the experiments with increasing system size exposed a practical limitation of the direct implantation. Namely, as the system size increased, the simulation time increased significantly, suggesting that this implementation may be impractical for a system with a high number of dimensions, which is the case for the models described in Chapter 2. The next sections detail this computational cost and how we overcome it.

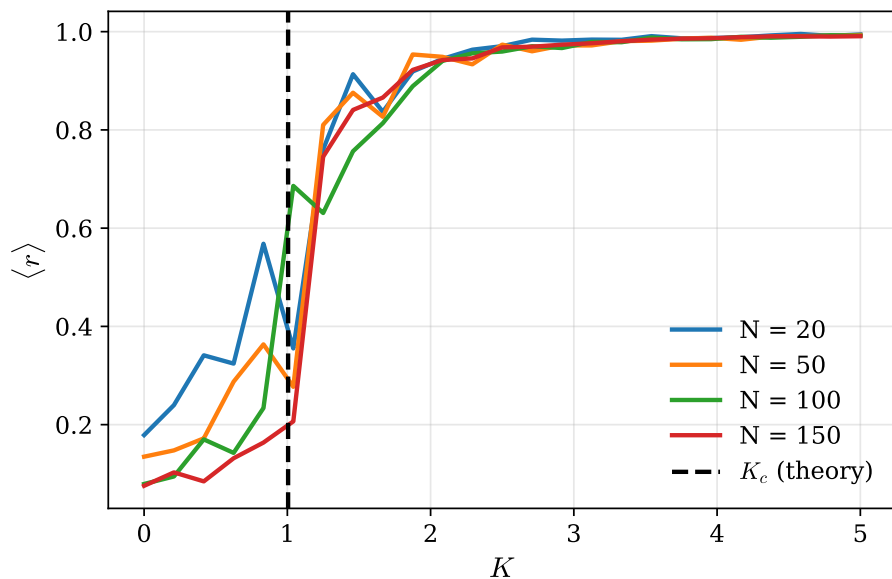


Figure 3.1: The steady-state order parameter $\langle r \rangle$ plotted as a function of K for $N = 20, 50, 100$, and 150 , with the theoretical K_c indicated.

3.1.4 Computational Cost of the Direct Implementation

Although our direct implementation reproduces the expected dynamics, it has a substantial computational cost. The double loop over oscillators requires evaluation of N^2 pairwise interactions when the Kuramoto model function is called. Hence, the system's computational complexity scales as $\mathcal{O}(N^2)$.

To quantify this scaling, we measured the runtime as a function of the size of the system. Figure 3.2 shows the average simulation time as a function of N on a log-log

scale. The slope of the resulting curve is approximately 2, demonstrating quadratic scaling.

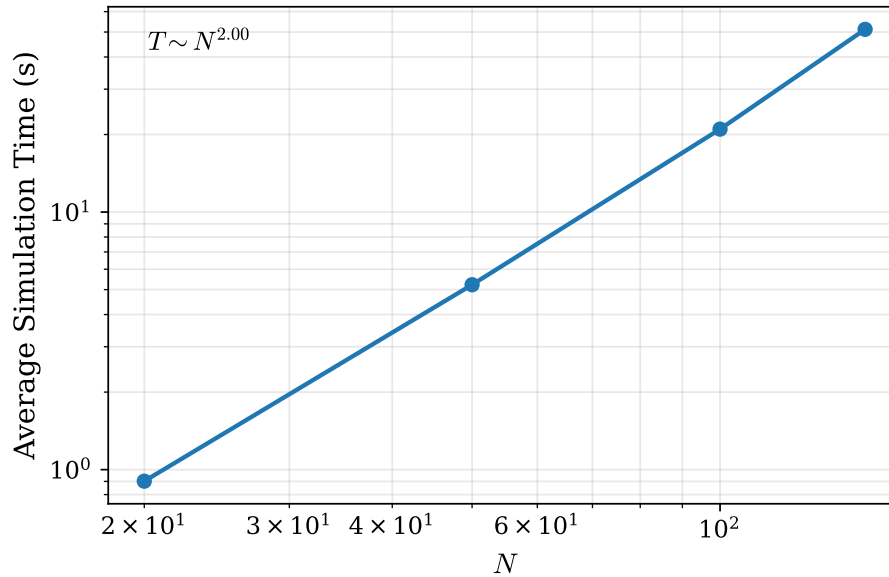


Figure 3.2: Log-log plot of the average runtime of the direct (pure Python) Kuramoto implementation as a function of system size N . The approximately linear trend indicates power-law scaling, with $T \sim N^2$.

Although this scaling is acceptable for small systems, it becomes prohibitive for larger networks or extensive parameter sweeps. Since our subsequent simulations are for systems with larger dimensions, computational efficiency becomes a limiting factor.

3.1.5 Introducing Just-In-Time Compilation

Just-In-Time (JIT) compilation was introduced to overcome the computational cost scaling inherent to the model. Just-In-Time (JIT) compilation is a technique in which selected portions of code are compiled into optimized machine code at runtime, rather than being interpreted line-by-line. In Python, this can be achieved using the Numba library, which compiles numerically intensive functions, particularly those involving loops over arrays, into efficient native code.

We wrote a function that incorporates JIT for computing the right-hand side of Equation 1.2:

```

@njit
def kuramoto_rhs_numba(y, w, K, N):
    yd = np.zeros(N)
    for i in range(N):
        for j in range(N):
            yd[i] += K * np.sin(y[j] - y[i])
        yd[i] = w[i] + yd[i] / N
    return yd
def kuramoto_model_numba(t, y, w, K, N):
    return kuramoto_rhs_numba(y, w, K, N)

```

The function `kuramoto_rhs_numba` is decorated with `@njit` to enable Numba to compile the function into optimized machine code at runtime. Compiling the function into machine code at run time ensures that the pairwise interaction calculations are executed at near-native speed, significantly reducing runtime compared to the pure Python implementation.

A second function, `kuramoto_model_numba`, is defined as a wrapper for compatibility with the `solve_ivp` integrator. The integrator requires the right-hand side to accept time `t` as its first argument. Since the dynamics do not depend explicitly on time, this wrapper forwards the relevant arguments to the JIT-compiled function. This structure allows the computationally intensive portion of the model to be optimized independently, while preserving the required interface for numerical integration and maintaining consistency with the original formulation of the model.

This modification does not alter the governing equations, numerical integration method, or procedures used to compute observables. It only accelerates execution by compiling the nested loops into optimized machine code at runtime.

To determine whether the JIT-compiled implementation improved computational efficiency, we repeated the runtime scaling experiment using the JIT-compiled implementation and compared the runtime curves of the pure Python and JIT-accelerated versions. The comparison is visualised in Figure 3.3.

The figure shows that the average runtime of integration plotted as a function of the system size at log scale. It also shows that the two curves have a vertical separation and different slopes. This behaviour represents two differences in the implementations. First, the vertical separation shows that the JIT-compiled version has a smaller scaling prefactor, C , in the scaling relation $T(N) \sim CN^\alpha$. Further, the smaller slope of the

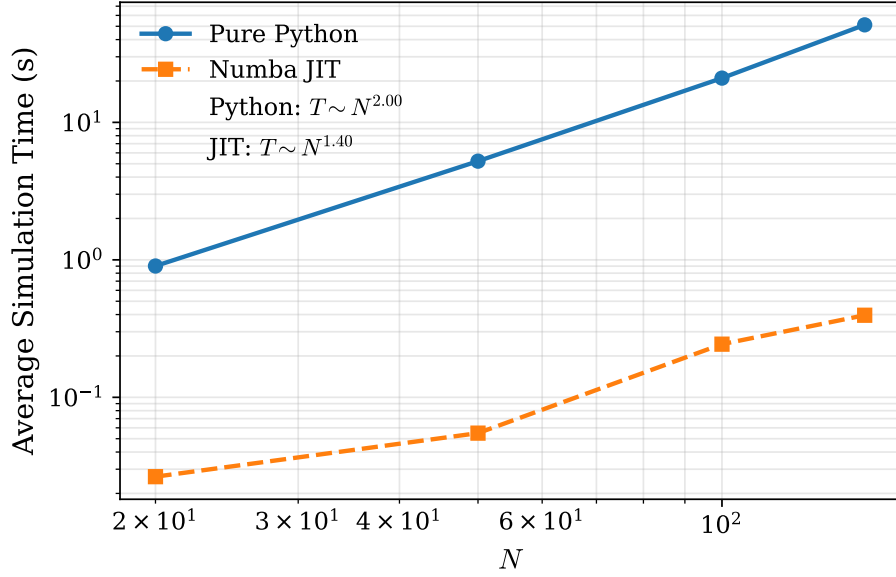


Figure 3.3: Log-log comparison of the average runtime plotted against the system size N for the pure Python and JIT-compiled implementations. The curves have different slopes, with the slope of the JIT-compiled function being smaller than that of the pure python implementation.

JIT-compiled curve indicates that the average running time of the JIT-compiled computation grows slower with system size compared to the pure Python implementation.

The combination of the downward shift and smaller slope of the log-log curve of the JIT-implementation compared to the pure python implementation suggests that JIT speeds up the computations. This speed up can be quantified using the speed up factor

$$S(N) = \frac{T_{Python}(N)}{T_{JIT}(N)}, \quad (3.3)$$

where $T_{Python}(N)$ and $T_{JIT}(N)$ are the simulation times of the pure-python and JIT-compiled implementations with system size N , respectively. The speed factor of the observed run times with system size is visualised in Figure 3.4.

Figure 3.4 shows that the speed up factor increases with increasing system size. This increase in the speed-up factor with system size is because the computation time of computations with the JIT-compiled function grows slower with system size compared to the computations with the pure python function.

This subsection shows that JIT compilation can accelerate the implementation time

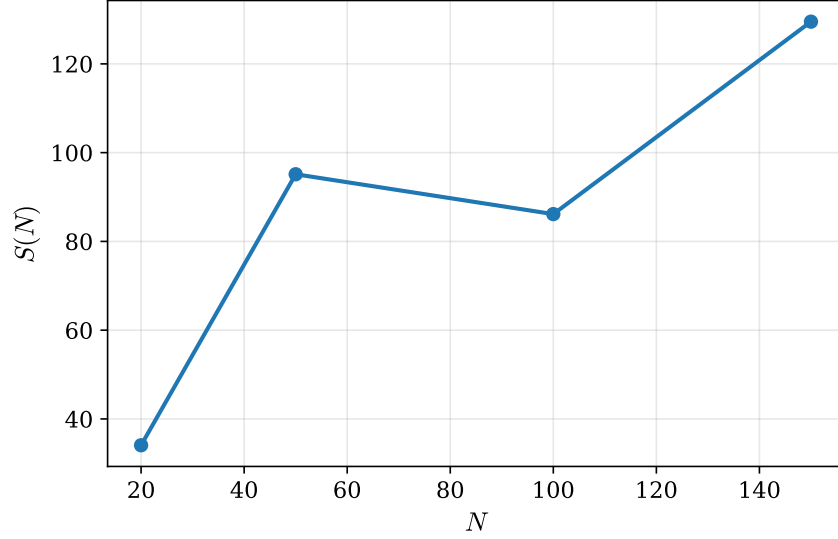


Figure 3.4: Speed-up factor $S(N)$ versus the system size N . The speed up factor increases with system size, showing that the effects of JIT compilation are more useful for larger systems.

of the Kuramoto model without changing its dynamics. In the next subsection, we show that JIT compilation can be applied to simulate a larger system that better demonstrates the transition to synchrony.

3.1.6 Large System Onset of Synchrony Experiment with Numba JIT

To conclude the section, we demonstrate that the JIT-accelerated implementation enables simulations at system sizes beyond those that are practical with the pure python version of our function. Specifically, we compute the onset-of-synchrony curve for $N = 1000$ using the JIT function, implementing the simulation with the absolute and relative tolerances equal to $1e - 12$, $T = 50$, and $dt = 0.1$. The results of this experiment are presented in Figure 3.5.

The onset of synchrony curve in Figure 3.5 has the same qualitative features observed for the systems with smaller N but with a sharper transition. In addition, the figure shows that the wall-clock runtime required to compute the curve is ~ 1100 s. This run time is manageable for a system of $N = 1000$. Hence, the results highlight the practical feasibility gained through JIT compilation.

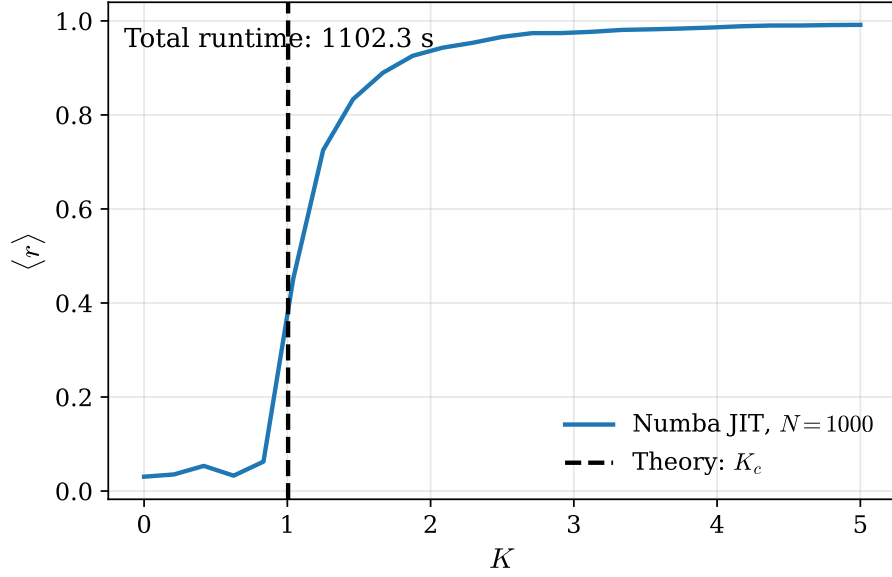


Figure 3.5: Steady-state order parameter $\langle r \rangle$ as a function of the coupling strength K for a large system ($N = 1000$) computed using the JIT-compiled Kuramoto model python function. The theoretical critical coupling K_c is indicated by a vertical dashed line, and the total wall-clock time required to generate the curve is reported on the figure.

The above result serves two purposes. First, it reinforces that the JIT implementation preserves the expected Kuramoto dynamics at larger scales. Second, it provides direct evidence that the computational acceleration is not merely incremental but enables the execution of large- simulations within reasonable runtime, which will be required for the computational demands of later chapters.

This section reported the development of a direct numerical implementation of the Kuramoto model. This implementation was validated it by reproducing the theoretical critical coupling strength and the sharpening of the synchronization transition with increasing system size N . We then identified a computational bottleneck inherent to the all-to-all coupling structure, verifying it using runtime scaling for the pure-python implementation. This motivated the introduction of JIT compilation, which accelerates the same governing equations without changing the model or numerical method. The improvement was demonstrated through scaling and speed-up analyses, and further illustrated by a large-scale onset-of-synchrony.

3.2 Developing a Python Function for Simulating the Kuramoto–Sakaguchi Model

This section discusses how we altered the function in Section 3.1. to incorporate the frustration term introduced by Sakaguchi in 1986 [82]. For computational efficiency, we adopted the JIT-version of the script. As stated earlier, the frustration term was introduced to capture unavoidable phase shifts in synchronization phenomena and to model repulsive or inhibitory interactions between oscillators [64, 81, 82]. We transcribed the Kuramoto–Sakaguchi model, Equation (1.5), as follows:

```
@njit
def kuramoto_rhs_numba(y, w, v, N):
    yd = np.zeros(N)
    for i in range(N):
        for j in range(N):
            yd[i] += v(0) * np.sin(y[j] - y[i] - v(1))
        yd[i] = w[i] + yd[i] / N
    return yd
def kuramoto_model_numba(t, y, w, v, N):
    return kuramoto_rhs_numba(y, w, v, N)
```

This function differs from the Kuramoto model function in one way: a column vector v that stores the coupling and frustration parameters is introduced. The vector takes the form $v = [K, \alpha]$. Hence, $v[0]$ corresponds to the coupling parameter, while $v[1]$ corresponds to the frustration parameter.

We tested the accuracy of this transcription in a manner similar to that used to test the validity of the Kuramoto model transcription. In the literature, analytical solutions for $N \rightarrow \infty$ show that $K_c(\alpha) = K_c(0)/\cos(\alpha)$ for a system where $g(\omega)$ follows a unimodal distribution that is symmetric about the mean [64]. Furthermore, the Kuramoto–Sakaguchi model exhibits synchronisation only for $|\alpha| < \pi/2$. Outside this region, various states, such as travelling, clustering, and turbulence, arise [64]. The periodicity of the sin function, which determines the degree of attraction or repulsion between oscillators, ensures that these regions are reproduced at intervals of 2π , such that $|\alpha| < \pi/2$ defines the principal synchronisation region. As in Section 3.1, we tested the validity of the function using onset-of-synchrony experiments.

For the experiment, we used the implementation functions in Section 3.1.3, where the absolute and relative tolerances were $1e-12$, $T = 50$, and $dt = 0.1$. We selected the natural frequencies of the oscillators from a Gaussian distribution with mean $\mu = 0$ and standard deviation $\sigma = 0.63$, as in the previous section. We selected frustration values of $\alpha = 0.31, 0.63, 0.94$, and 1.26 . These parameters produce $K_c \approx 1.06, 1.24, 1.71$, and 3.25 , respectively. The results of the onset-of-synchrony experiments are given in Figure 3.6.

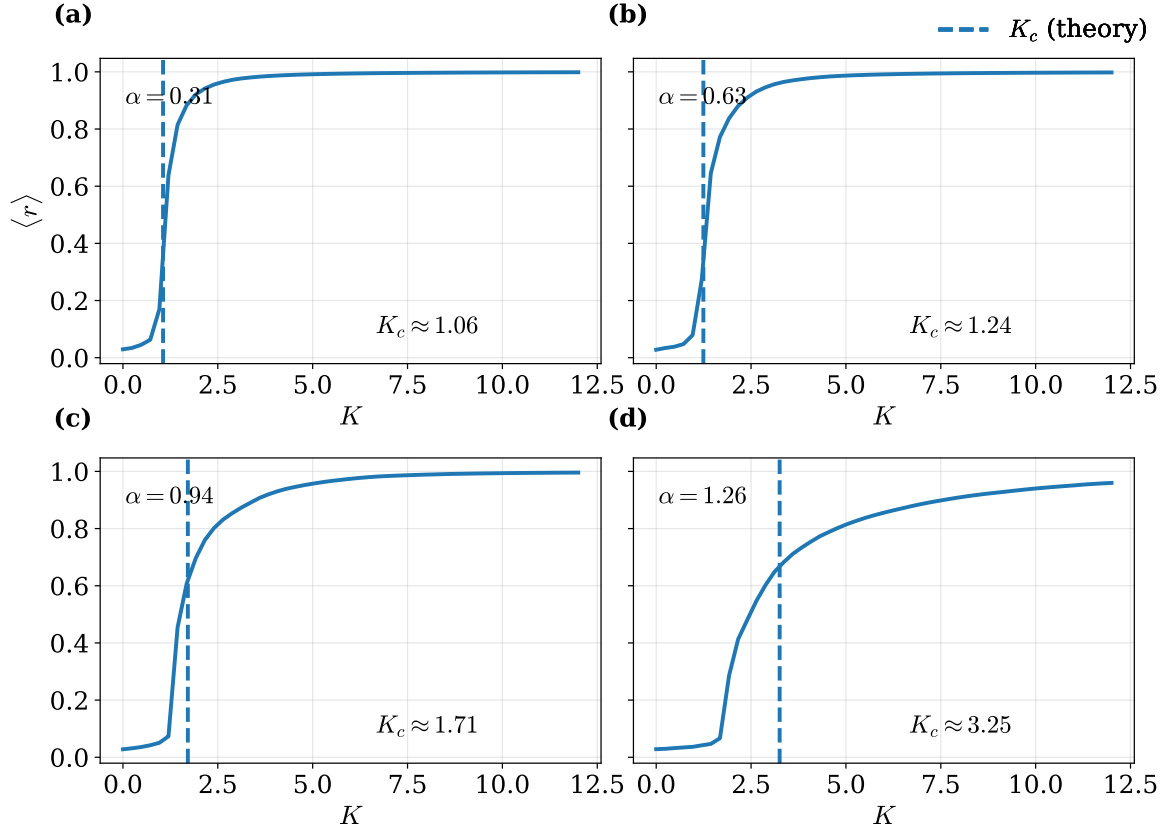


Figure 3.6: Steady-state order parameter $\langle r \rangle$ as a function of coupling strength K for the Kuramoto-Sakaguchi model with $\alpha =$ (a) 0.31, (b) 0.63, (c) 0.94, and (d) 1.26, for a system with $N = 1000$ computed using a JIT-compiled function. The theoretical critical coupling K_c in each panel is indicated by a vertical dashed line.

Figure 3.6 shows the steady-state average order parameter $\langle r \rangle$ as a function of the coupling strength K for frustrations $\alpha = 0.31, 0.63, 0.94$, and 1.26 . We selected frustrations in the range $(0, \pi/2)$, where the critical coupling K_c increases with α . This behaviour is observed in the figure, with the onset of synchrony shifting to larger K as α increases.

Before the critical coupling point K_c , a gradual but small increase in $\langle r \rangle$ is observed. This behaviour is due to finite-size and finite-time effects, where partial clustering of oscillators can occur even below the theoretical transition point [64, 81]. As K increases, a more rapid increase in $\langle r \rangle$ is observed, corresponding to the system-wide onset of synchrony. This transition becomes less sharp for larger values of α , reflecting the inhibiting effect of frustration on synchronisation.

These numerical results show good qualitative agreement with theoretical predictions, with deviations near the transition attributable to the finite size of the system and the finite integration time used in the simulations. This agreement shows that the implementation employed here accurately models the Kuramoto–Sakaguchi model.

Now that we have developed a function that fully simulates the Kuramoto and Kuramoto–Sakaguchi models, we can begin introducing synaptic coupling between oscillators. The subsequent section reports on the development of scripts that simulate models with STDP.

3.3 Developing a Python Function for Simulating the Kuramoto–Sakaguchi Model with STDP

As mentioned in the Introduction and Literature Review, the Kuramoto model with plasticity was first presented by [95] in 2002. In this model, plasticity was represented by a cosine function [95]. The next formulation was proposed by Maistrenko et al. in 2007 [94]. Instead of using a cosine function, they used STDP to model changes in the synaptic weights between oscillators [94]. Here, we attempt to reproduce Maistrenko et al.’s results as part of building a script for the Kuramoto model with STDP.

For the Kuramoto model with all-to-all coupling, introducing STDP adds $N(N - 1)$ additional equations, significantly increasing the computational cost of the model. Such computational demands remain challenging to simulate within reasonable time. Hence, we incorporate numba JIT in the subsequent simulations.

We augmented the function in Section 3.1 to account for STDP. The script for this new function is as follows:

```
@njit
def kuramoto_stdp_rhs_numba(y, w, v, N):
```

```

yd = np.zeros(N+N*N)
k = N
for i in range(N):
    for j in range(N):
        theta = y[j] - y[i]
        theta_wrap = (theta + np.pi)%(2*np.pi) - np.pi

        if theta_wrap >= 0:
            yd[k] = v[2] * (v[0] - y[k]) * exp(-theta/v[3])
        else:
            yd[k] = -v[2] * y[k] * exp(theta/v[4])

        yd[i] += yd[k] * np.sin(theta-v[1])
        k += 1
    yd[i] = w[i] + yd[i] / float(N)

return yd

def kuramoto_stdp_model_numba(t, y, w, v, N):
    return kuramoto_stdp_rhs_numba(y, w, v, N)

```

In the function, y is a column vector that stores the phases of the oscillators and the coupling strengths between them. The first N entries correspond to the oscillator phases, and the last N^2 entries correspond to the connection strengths. yd stores the instantaneous frequencies in the first N entries and the instantaneous changes in the synaptic strengths in the last N^2 entries. w stores the natural frequencies of the oscillators. v is now a larger column vector of the form $\mathbf{v} = [A, \alpha, \epsilon, \tau_p, \tau_d]^T$, where A is the maximum coupling between neurons, α is the Kuramoto–Sakaguchi frustration parameter, ϵ is the learning rate, and τ_p and τ_d are the potentiation and depression time constants, respectively. Maistrenko et al. did not consider frustration in their model [94]. Hence, we set $\alpha = 0$. N is still the number of neurons in the system.

In the function, k is an index for y and yd that corresponds to entries related to the synaptic strengths. It starts at $k = N$ because the first N entries of both vectors correspond to the oscillator phases and frequencies, respectively. In the inner loop, the phase difference between oscillators, θ , is wrapped to the range $(-\pi, \pi]$ using modulo arithmetic. Wrapping it to this range allows us to use it in the STDP rule. The `if` statements use the wrapped phase difference to determine whether potentiation or depression occurs. When the wrapped difference is greater than or equal to zero, poten-

tiation occurs; when it is less than zero, depression occurs. Within the `if` statements, the potentiation and depression are computed according to Equation 2.1.

Maistrenko et al. [94] showed that the Kuramoto model with STDP coupling exhibits multistability when $\tau_p < \tau_d$. They assumed $\tau_d > \tau_p$. Here, we test the validity of the above function by reproducing this result for $N = 2$. In their experiment, Maistrenko et al. [94] compared the hysteresis loop behaviour for systems with $\tau_d = \tau_p = 0.3$ and $\tau_p = 0.15$ and $\tau_d = 0.3$. They varied A from 0 to 4, and then from 4 to 0. The parameters of these systems were set as $N = 2$ and $\epsilon = 0.5$, and the natural frequencies of the oscillators were set to 1 and 2. We reproduced this experiment with identical parameters and initial conditions by computing the function presented above. The results of this experiment are shown in Figure 3.7.

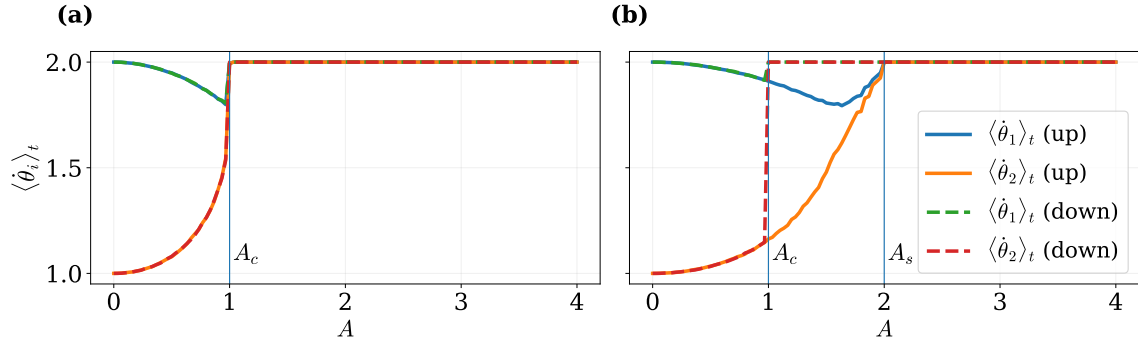


Figure 3.7: Time averaged running frequencies of oscillators $i = 1$ and 2 for (a) $\tau_d = \tau_p = 0.3$ and (b) $\tau_p = 0.15$ and $\tau_d = 0.3$ as a function of A . The figure shows the curves when A is increased and decreased. A_C is the maximum coupling strength that causes synchronisation for $\tau_d = \tau_p = 0.3$ and A_S is the maximum coupling that causes synchronisation for $\tau_p = 0.15$ and $\tau_d = 0.3$ with increasing A . The figure shows that for $\tau_p = 0.15$ and $\tau_d = 0.3$ the range (A_C, A_S) has the coexistence of desynchronised and synchronised states.

Figure 3.7 shows the change in the running average frequencies of two oscillators as the maximum STDP coupling is varied. The figure shows that sweeping A up and down does not produce hysteresis for $\tau_p = \tau_d$. In both directions, the transition from the desynchronised state to the synchronised state occurs at A_C . When asymmetry is introduced, that is, $\tau_d = 0.3$ and $\tau_p = 0.15$, increasing and then decreasing A forms a hysteresis loop. When A is increased, synchronisation occurs at A_S , and when A is decreased, desynchronisation occurs at A_C . This creates the interval (A_C, A_S) where the synchronised and desynchronised states coexist. This behaviour is called multistability. Our ability to reproduce this multistability confirms that our modelling function is

accurate.

Now that we have established a working function for the Kuramoto–Sakaguchi model with STDP coupling, we must develop a function that can account for different forms of coupling. The first form of coupling we investigate is cyclic coupling. The next section discusses the development of this function.

3.4 Developing Python Functions for Simulating the Kuramoto–Sakaguchi Model with STDP and Cyclic Coupling

Ansariara et al. [104] introduced cyclic connectivity in the model developed by Maistrenko et al. [94]. They also introduced frustration to the model while maintaining symmetry in the potentiation and depression time constants [104]. This section reproduces their results in an attempt to create a function that can simulate models with different connectivity structures between neurons.

First, we developed a function that creates the adjacency matrix of the neurons. It ensures that each neuron is connected to n other oscillators, with $n/2$ connections to its left and $n/2$ to its right. The function for creating this matrix is given below:

```
def cyclic_adjacency_matrix(N, n):
    a = np.zeros((N, N), 'd')
    m = int(n / 2)
    for i in range(N):
        for j in range(i - m, i + m + 1):
            if i != j:
                a[i, mod(j, N)] = 1
    return a
```

In the function, N is the number of neurons and n is the node degree of each neuron. The function creates an $N \times N$ matrix, where the row indices represent the presynaptic neurons and the column indices represent the postsynaptic neurons. We denote this matrix by G . When $G_{ij} = 1$, the presynaptic neuron i has a synapse with the postsynaptic neuron j . When an entry is zero, the neurons do not have a connection.

The first loop runs over all neurons to account for each neuron in the system. The inner loop ranges from $i - n/2$ to $i + n/2$, ensuring that each neuron i is connected to its nearest n neighbours. The condition $i \neq j$ ensures that neurons are not connected to themselves. The loop sets entries corresponding to connections from 0 to 1. The `mod` function uses modulo arithmetic to enforce cyclic boundary conditions, where neurons near the boundaries are connected to neurons at the opposite end of the system. An example of the adjacency matrix produced by this function is shown in Equation 2.7 for $N = 12$ and $n = 6$. Using this matrix in the function developed in Section 3.3 adds unnecessary overhead by considering zero entries where there are no connections. Hence, we a function that converts the adjacency matrix into an indexing matrix:

```
def indexing_matrix(a):
    n = int(len(a[:, 0]))
    p = int(a[0].sum())
    im = np.zeros((n, int(p)), 'i')
    for i in range(n):
        k = 0
        for j in range(n):
            if a[i, j] == 1:
                im[i, k] = j
                k += 1
    return im
```

This function creates an $N \times n$ matrix where the entries of the i -th row store the indices of postsynaptic neurons that have a connection with presynaptic neuron i . The function takes an $N \times N$ adjacency matrix as an argument. This matrix can be the cyclic adjacency matrix above or can be an adjacency matrix of an other topology where each node has the same number of connections. The function first reads the number of rows of the inputted adjacency matrix. Then, it determines the number of connections that each neuron has by summing the entries of the first row. This method works because each neuron has the same number of connections as any other neuron for the cyclic network matrix. Then, an $N \times n$ matrix of zeros is set up. The two loops fill this matrix. The if statement in the inner loop is used to fills the new matrix with the indices of columns where the entry G_{ij} is one. An example of the indexing matrix, which corresponds to the adjacency matrix in Equation 2.7 is given below:

$$I = \begin{pmatrix} 1 & 2 & 3 & 9 & 10 & 11 \\ 0 & 2 & 3 & 4 & 10 & 11 \\ 0 & 1 & 3 & 4 & 5 & 11 \\ 0 & 1 & 2 & 4 & 5 & 6 \\ 1 & 2 & 3 & 5 & 6 & 7 \\ 2 & 3 & 4 & 6 & 7 & 8 \\ 3 & 4 & 5 & 7 & 8 & 9 \\ 4 & 5 & 6 & 8 & 9 & 10 \\ 5 & 6 & 7 & 9 & 10 & 11 \\ 0 & 6 & 7 & 8 & 10 & 11 \\ 0 & 1 & 7 & 8 & 9 & 11 \\ 0 & 1 & 2 & 8 & 9 & 10 \end{pmatrix}. \quad (3.4)$$

Now, that we have set up the adjacency matrix function and its accompanying indexing matrix, we can incorporate the latter to the function of the Kuramoto–Sakaguchi model with STDP coupling presented in Section 3.3.

The new function for the Kuramoto model with STDP that accounts for varying the adjacency matrix is as follows:

```
@njit
def kuramoto_stdp_coupling_rhs_numba(y, w, v, ii, N, n):
    yd = np.zeros(N*(n+1))
    k = N
    for i in range(N):
        for j in range(n):
            theta = y[ii[i,j]] - y[i]
            theta_wrap = (theta + np.pi)%(2*np.pi) - np.pi

            if theta_wrap >= 0:
                yd[k] = v[2] * (v[0] - y[k]) * exp(-theta/v[3])
            else:
                yd[k] = -v[2] * y[k] * exp(theta/v[4])

            yd[i] += yd[k] * np.sin(theta-v[1])
            k += 1
    yd[i] = w[i] + yd[i] / float(n)
```

```

    return yd
def kuramoto_stdp_coupling_model_numba(t, y, w, ii, v, N, n):
    return kuramoto_stdp_coupling_rhs_numba(y, w, ii, v, N, n)

```

This function is almost identical to that in Section 3.3. The difference is that we introduce the argument `ii`, which is the indexing matrix described in the previous paragraph. Furthermore, we include the argument `n`, which is the number of connections that each neuron has. Introducing coupling that is not all-to-all reduces the number of equations in the model from N^2 to $N(n+1)$. This is because we now have Nn equations for the dynamics of the coupling strengths and the original N equations for the dynamics of the neuron frequencies. Hence, `yd` becomes a column vector with $N(n+1)$ rows. The index k , which tracks the coupling dynamics, still starts at N , but now counts up to $k = N + n$ for each neuron. The indexing of the outer loop remains unchanged because it follows the indices of the neurons. The inner loop now runs up to n because each neuron has n connections. Entries corresponding to non-existent connections are not considered.

To confirm the correctness of the function, we reproduced results presented by Ansariara et al. [104] and Botha et al. [69]. Ansariara et al. [104] found that a model with cyclic connectivity between neurons and a small frustration produces a phase-locked state. They used the parameters $\tau_p = \tau_d = 0.05\pi$, $A = 1.000$, $\epsilon = 0.1$, and $\alpha = 0.015\pi$ with $N = 101$ and $n = 50$. We performed an experiment with identical parameters and obtained the following results:

We reproduced the result observed by Ansariara et al. [104]. As shown in the above figure, the phases reach a phase synchronized state with leading and lagging neurons. As observed by Ansariara et al., the leading neuron has a blank row in the connectivity matrix. This shows that it is no longer the pre-synaptic neuron of any synapse. Hence, its dynamics are no longer influenced by the other neurons in the system. However, it remains the post-synaptic neuron of all the neurons it was initially connected to, indicated by its column connections being equal to one. This means that it influences the other neurons while not being influenced by them. This behaviour meets the definition of a pacemaker [104]. This pacemaker behaviour was observed by Ansariara et al. [104], indicating the validity of our function.

Although we have confirmed the validity of our function by reproducing the results of Ansariara et al. [104], we also reproduce a result presented by Botha et al. [69] for comparison with our results. We show that for $\tau_p = \tau_d = 0.003$, $\epsilon = 0.01$, $A = 1.0$,

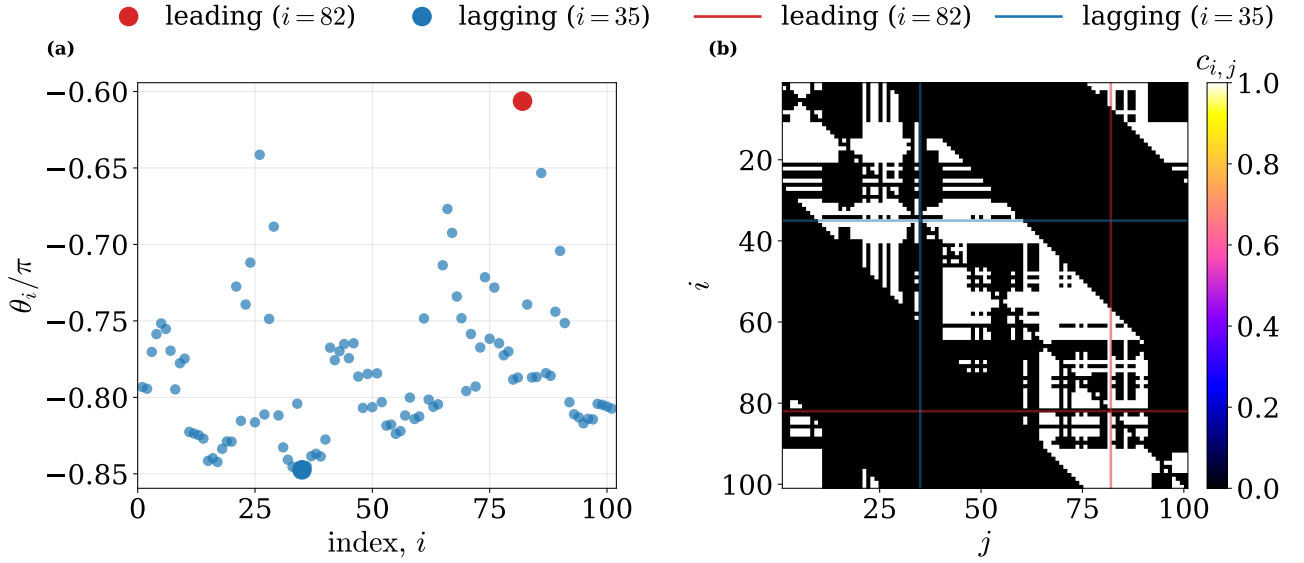


Figure 3.8: Steady-state (a) neuron phases and (b) interneuron coupling strengths for $N = 101, n, \tau_p = \tau_d = 0.05\pi, A = 1.000, \epsilon = 0.1$, and $\alpha = 0.015\pi$. In (a), there is a leading neuron marked in the larger red dot and lagging neuron marked with a larger blue dot. In (b), the rows and columns of the leading and lagging neurons are indicated in corresponding colours. In this panel, along the leading oscillator row there are no connections and the original connections along the corresponding column are retained. The opposite is true for the lagging neuron.

and $\alpha = 1.478$, a shifting chimera state emerges, as observed by Botha et al. [69]. Our reproduction is shown below:

Figure 3.9 shows our reproduced result of the chimera state observed by Botha et al. [69] for a cyclic network with the parameters stated above. This chimera state consists of neurons with entrained frequencies alongside neurons that are not entrained. The set of entrained neurons changes over time, making this a moving chimera state. This chimera state will be compared to the results for a random network in Chapter 4.

Figure 3.10 shows the behaviour of the Kuramoto–Sakaguchi model with STDP under nonlocal cyclic coupling. It shows that, for most of the regions $0 < \alpha < \pi/2$ and $3\pi/2 < \alpha < 2\pi$, the system is fully synchronised (purple regions), i.e. all oscillators have the same frequency and identical phases for any given α . In the fully synchronised regions, the order parameter is 1.

For $\pi/2 < \alpha < 3\pi/2$, the system becomes phase-synchronised (white regions), i.e. the oscillators have the same instantaneous frequencies but their phases are distributed

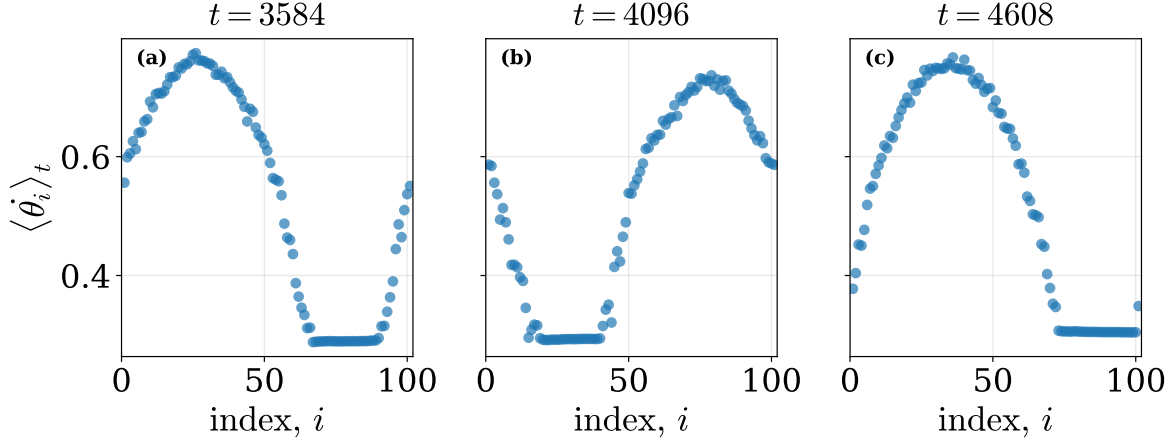


Figure 3.9: Running average frequencies at times (a) $t = 3584$, (b) $t = 4096$, and (c) $t = 4608$ for a system with $\tau_p = \tau_d = 0.003$, $\epsilon = 0.01$, $A = 1.0$, $\alpha = 1.478$, $N = 101$, and $n = 68$. The figure shows that for these parameters, some neurons have the same running average frequency, and others do not. This arrangement is called a chimera state. The chimera state is said to be non-stationary because the oscillators that have the same running average frequency change over time.

in a fixed pattern. Here, the order parameter is close to zero because the phases are widely spread. The red bars just before $\pi/2$ and just after $3\pi/2$ indicate regions where chimera states occur. These chimera states are of the same form as those presented in Figure 3.9.

Figure 3.10 is reproduced here for reference. It will be compared later to Figure 4.4, where the behaviour of the same system with random coupling is explored.

At this point, we have established working functions that can be used to simulate the behaviour of the Kuramoto–Sakaguchi model with STDP under cyclic coupling. In the next two sections, we present functions that can be used to simulate the model with random coupling and in a two-ensemble setting. These functions will allow us to investigate the behaviour of Eqs. 2.14 and 2.20.

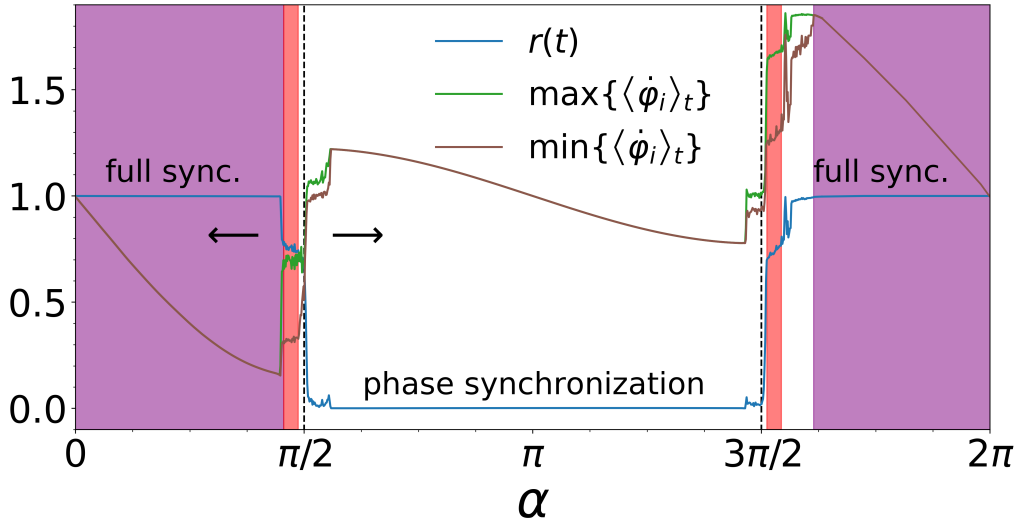


Figure 3.10: A reproduction of Figure 4A, from Botha et al. [69]. Order parameter and maximum and minimum running average frequencies as a functions of the frustration, α , for the system with cyclic coupling. The figure shows regions where the system is fully synchronised (puple), phase synchronised (white), and in a chimera states (pink), like the one seen in in Figure 3.9. Parameters: $N = 40$, $n = 28$, $\tau_p = 0.003$, $\tau_d = 0.004$, and $\omega_i = 1$.

3.5 Developing Python Functions for Simulating the Kuramoto–Sakaguchi Model with STDP and Random Coupling

One of our objectives is to extend the Kuramoto–Sakaguchi model with STDP to include random coupling between oscillators. Meeting this objective will allow us to investigate how random and sparse connectivity influences synchronisation dynamics and memory encoding, where we assume that the steady states of neuronal systems modelled by our framework represent memory-encoding ensembles. To realise these objectives, we must write a Python function that can simulate a Kuramoto–Sakaguchi system with STDP coupling on a random network. This section describes the development of such a function.

We have already established a working function for the Kuramoto–Sakaguchi model with STDP under an arbitrary coupling topology:

```
@njit
def kuramoto_stdp_coupling_rhs_numba(y, w, v, ii, N, n):
    yd = np.zeros(N*(n+1))
    k = N
    for i in range(N):
        for j in range(n):
            theta = y[ii[i,j]] - y[i]
            theta_wrap = (theta + np.pi)%(2*np.pi) - np.pi

            if theta_wrap >= 0:
                yd[k] = v[2] * (v[0] - y[k]) * exp(-theta_wrap/v[3])
            else:
                yd[k] = -v[2] * y[k] * exp(theta_wrap/v[4])

            yd[i] += y[k] * np.sin(theta-v[1])
            k += 1
        yd[i] = w[i] + yd[i] / float(n)

    return yd

def kuramoto_stdp_coupling_model_numba(t, y, w, ii, v, N, n):
```

```
return kuramoto_stdp_coupling_rhs_numba(y, w, ii, v, N, n)
```

What remains is the development of an appropriate helper function that produces the adjacency matrix of a random coupling network defined in Equation 2.8, where each oscillator i is equally likely to be connected to n other oscillators in a bidirectional manner for a system of size N . The following function creates such an adjacency matrix.:

```
def k_mat_ran(N, n):
    a = np.zeros((N, N), 'd')
    err = 1.0
    while err > 0.01:
        for i in range(N):
            j = randint(N)
            if j == i:
                a[i, j] = 0.0
            elif int(a[i].sum()) < nc:
                a[i, j], a[j, i] = 1.0, 1.0
            elif int(a[i].sum()) > nc:
                a[i, j], a[j, i] = 0.0, 0.0
        err = np.sum((a.sum(axis=1) - nc) ** 2)
    return a
```

The helper function `k_mat_ran(N,n)` generates an $N \times N$ symmetric adjacency matrix a corresponding to a random bidirectional coupling network in which each oscillator is connected to n other oscillators. It begins with an all-zero matrix and iteratively updates randomly selected entries a_{ij} while enforcing the absence of self-coupling ($a_{ii} = 0$). For each presynaptic neuron i , a candidate postsynaptic neuron j is chosen uniformly at random, and a connection is created or removed in a symmetric fashion ($a_{ij} \in \{0, 1\}$) depending on whether the current degree of neuron i is below or above the target value n .

The error term `err` monitors convergence. It is defined as the sum of the squared deviations of the row sums (i.e. the number of existing connections) from n . The iteration terminates once this quantity falls below a tolerance of 0.01, which was determined through trial and error. After termination, the resulting matrix a is returned as the desired random-coupling adjacency matrix. An example matrix produced by this function for $N = 12$ and $n = 6$ was presented in Equation (2.9). This matrix is then

passed to the indexing function presented in Section 3.4. Once the indexing matrix has been constructed, it can be used in the function `kuramoto_stdp_coupling_rhs_numba`. This workflow allows us to simulate the Kuramoto–Sakaguchi model with STDP under random coupling. The results of these simulations are presented in Chapter 4.

3.6 Developing Python Functions for the Two-Ensemble Kuramoto–Sakaguchi Model with STDP and Random Coupling

The function `kuramoto_stdp_coupling_model_numba` developed so far is for single-ensemble systems. The next task in our function development is to create a function that can be used to simulate a system with two ensembles. These ensembles should be randomly connected, and the inter-ensemble connections should also be random and bidirectional, following Equation 2.15. Our function should be able to simulate Equation 2.20.

We start by developing a function for the adjacency matrix of a two-ensemble system with random coupling and bidirectional connectivity. Below is a function we developed for this adjacency matrix:

```
def k_mat_ran_two_ensemble(N, n, b):

    a = np.zeros((2*N, 2*N), 'd')

    err = 1.0

    while err > 0.1:
        for i in range(2*N):
            if i < N:
                j = randint(N)
                while j == i:
                    j = randint(N)
            else:
                j = randint(N, 2*N)
                while j == i:
```

```

        j = randint(N, 2*N)

    if int(a[i].sum()) < n:
        a[i, j], a[j, i] = 1.0, 1.0
    elif int(a[i].sum()) > n:
        a[i, j], a[j, i] = 0.0, 0.0

    err = np.sum((a.sum(axis=1) - n) ** 2)

if b > 0:

    c = np.zeros((N, N), 'd')

    err = 1.0
    while err > 0.1:
        for i in range(N):
            j = randint(N)
            #if j == i:
            #    c[i, j] = 0.0
            if int(c[i].sum()) < b:
                c[i, j], c[j, i] = 1.0, 1.0
            elif int(c[i].sum()) > b:
                c[i, j], c[j, i] = 0.0, 0.0
            err = np.sum((c.sum(axis=1) - b) ** 2)

    a[0:N, N:2*N] = c
    a[N:2*N, 0:N] = c.transpose()

return a

return a

```

The function `k_mat_ran_two_ensemble(N, n, b)` produces a $2N \times 2N$ matrix `a` that represents the adjacency of the two-ensemble network. The matrix is initialised as an array of zeros and populated systematically. First, the logic used to create a single ensemble with random connectivity is reused. The tolerance `err` is used as in `k_mat_ran(N, n)` to ensure that the intra-ensemble connectivity degree of each neuron

is n . The first **if** statement in the **for** loop ensures that neurons in the first ensemble are connected to neurons in the same ensemble by enforcing $j < N$. The **else** statement ensures that for $i \geq N$, bidirectional connections are formed between oscillators in the second ensemble. This block of the function populates the two diagonal blocks $G^{(1)}$ and $G^{(2)}$ in Equation 2.15.

The second block of the function, beginning with the statement **if b>0**, populates the off-diagonal blocks of Equation 2.15. We define an $N \times N$ matrix and populate it as a symmetric single-ensemble adjacency matrix. After this matrix is constructed, it is used to populate the off-diagonal blocks of the main matrix: the top-right block is set equal to it, and the bottom-left block is set equal to its transpose. This operation ensures that $G^{(2,1)} = G^{(1,2)^T}$, as described in Section 2.6, ensuring bidirectional coupling between the ensembles. The matrix **c** is made symmetric to ensure that it has the same number of ones row- and column-wise. This, in turn, ensures that the rows of the top-right and bottom-left blocks have the same number of ones. Hence, the inter-ensemble coupling degree of b is maintained for both ensembles. For a system with $N = 6$, $n = 3$, and $b = 3$, the function produces, for example, the following adjacency matrix:

$$\mathcal{A} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}. \quad (3.5)$$

In the matrix, the first six columns of each of the first six rows have at most three ones since $n = 3$. This is also the case for the last six columns of the last six rows. This is expected because these blocks represent the two diagonal blocks. Each of these blocks is symmetric because of the bidirectionality condition. The last six columns of the first six rows have a maximum of three ones as well because $b = 3$, and the bottom-left block is the transpose of the top-right block because of the bidirectionality condition. Hence, the function outputs the required adjacency matrix.

As in Section 3.4, we do not use this adjacency matrix directly. Instead, we create

an indexing matrix using the adjacency matrix function below:

```
def ind_two_ensemble(a):
    n = int(len(a[:, 0]))
    p = int(a[0].sum())
    m = int(n/2)
    im = np.zeros((n, p), 'i')
    for i in range(m):
        k = 0
        for j in range(n):
            if a[i, j] == 1:
                im[i, k] = j
                k += 1

    for i in range(m, n):
        k = 0
        for j in range(m, n):
            if a[i, j] == 1:
                im[i, k] = j
                k += 1
        for j in range(m):
            if a[i, j] == 1:
                im[i, k] = j
                k += 1
    return im
```

. The difference this time is that the matrix has dimensions $2N \times (n + b)$. The rows are still linked to presynaptic neurons, while the entries represent indices of postsynaptic neurons connected to the presynaptic neuron of that row. The number of columns accounts for the n intra-ensemble couplings and b inter-ensemble couplings per neuron i . The first n columns of each row correspond to the postsynaptic neuron indices for the intra-ensemble couplings, and the last b columns correspond to postsynaptic neuron indices for the inter-ensemble couplings. For example, the indexing matrix of

the adjacency matrix in Equation 3.5 is

$$I = \begin{pmatrix} 2 & 4 & 5 & 8 & 9 & 10 \\ 2 & 4 & 5 & 7 & 9 & 10 \\ 0 & 1 & 3 & 6 & 9 & 11 \\ 2 & 4 & 5 & 6 & 7 & 8 \\ 0 & 1 & 3 & 6 & 7 & 11 \\ 0 & 1 & 3 & 8 & 10 & 11 \\ 8 & 9 & 10 & 2 & 3 & 4 \\ 8 & 9 & 11 & 1 & 3 & 4 \\ 6 & 7 & 11 & 0 & 3 & 5 \\ 6 & 7 & 10 & 0 & 1 & 2 \\ 6 & 9 & 11 & 0 & 1 & 5 \\ 7 & 8 & 10 & 2 & 4 & 5 \end{pmatrix} \quad (3.6)$$

and follows the expected pattern. For example, for the first row, the first three columns have entries with values less than $N = 6$, indicating that all indexed postsynaptic neurons are in the first ensemble. The entries in the final three columns are greater than or equal to $N = 6$, indicating that they represent indices of postsynaptic neurons in the second ensemble. From the seventh row onward, the first three entries are greater than or equal to $N = 6$ and the last three entries are less than $N = 6$. This indicates that the second half of the rows corresponds to presynaptic neurons in the second ensemble.

Now that we have set up the adjacency matrix and its corresponding indexing matrix, we must develop a function that transcribes Equation 2.20. This function is presented below:

```
@njit
def kuramoto_stdp_coupling_two_ensemble_rhs_numba(y, w, v, ii, N, n, b):
    yd = np.zeros(2*N*(n+b+1))
    R = 2*N
    m = n + b
    k = R
    for i in range(R):
        w_1 = 0.0
        w_2 = 0.0

        for j in range(n):
```

```

theta = y[ii[i, j]] - y[i]
theta_wrap = (theta + np.pi) % (2*np.pi) - np.pi

if theta_wrap >= 0:
    yd[k] = v[2] * (v[0] - y[k]) * exp(-theta / v[3])
else:
    yd[k] = -v[2] * y[k] * exp(theta / v[4])

w_1 += y[k] * np.sin(theta - v[1])
k += 1

for j in range(n, m):
    theta = y[ii[i, j]] - y[i]
    theta_wrap = (theta + np.pi) % (2*np.pi) - np.pi

    if theta_wrap >= 0:
        yd[k] = v[2] * (v[0] - y[k]) * exp(-theta / v[3])
    else:
        yd[k] = -v[2] * y[k] * exp(theta / v[4])

    w_2 += y[k] * np.sin(theta - v[1])
    k += 1

yd[i] = w[i] + w_1 / float(n) + (w_2 / float(b) if b > 0 else 0.0)

return yd

def kuramoto_stdp_coupling_model_two_ensemble_numba(t, y, w, v, ii, N, n, b):
    return kuramoto_stdp_coupling_two_ensemble_rhs_numba(y, w, v, ii, N, n, b)

```

Compared to the function introduced in Section 3.4, this function still takes the arguments y , w , ii , v , N , n . However, some of these arguments are slightly different. N and n are now the size of a single ensemble and the number of intra-ensemble connections each oscillator has, respectively. w is still a vector that stores the natural frequencies of the oscillators; however, instead of being an N -dimensional vector, it is now a $2N$ -dimensional vector. ii is an indexing matrix as described for the function `ind_two_ensemble(a)`. It has shape $2N \times (n+b)$, where n and b are the intra- and inter-

ensemble connections per oscillator, respectively. $\mathbf{y}$ is now a $2N(n+b+1)$ -dimensional vector that stores the oscillator phases in its first $2N$ entries and the pairwise coupling strengths in the last $2N(n+b)$ entries. The new argument $\mathbf{b}$ is the number of inter-ensemble connections per oscillator. The vector $\mathbf{v}$ is as defined in Section 3.4; it stores the maximum coupling A , frustration α , learning rate ϵ , and the potentiation and depression time constants τ_p and τ_d , respectively. We assume that the parameters for intra-ensemble coupling are the same as those for inter-ensemble coupling. This assumption is made for simplicity and is not intended as a biophysically grounded constraint.

In the function, $\mathbf{y}\mathbf{d}$ is a column vector that stores the instantaneous frequencies of the oscillators in the first $2N$ entries and the instantaneous changes in synaptic weights in the last $2N(n+b)$ entries. $\mathbf{R}$ is the total number of oscillators and $\mathbf{m}$ is the total number of connections per oscillator. The index $\mathbf{k}$ is used to point to coupling-strength positions in the $\mathbf{y}$ and $\mathbf{y}\mathbf{d}$ vectors. The outer loop runs through all oscillators in the system. Within this loop, $\mathbf{w_1}$ and $\mathbf{w_2}$ accumulate the intra- and inter-ensemble coupling contributions, respectively, to oscillator i . The first inner loop accounts for intra-ensemble couplings, while the second inner loop accounts for inter-ensemble couplings. After both loops are completed, the cumulative intra- and inter-ensemble effects are scaled by n and b , respectively, and added to ω_i to determine the instantaneous frequency of oscillator i . When $b = 0$, the inter-ensemble contribution is eliminated and the two ensembles evolve independently.

We have now established three functions. The first generates an adjacency matrix for a two-ensemble system with random bidirectional connections. The bidirectionality condition is imposed to simplify the implementation and maintain consistent data structures. The second function constructs an indexing matrix that converts the $2N \times 2N$ adjacency matrix into a $2N \times (n+b)$ indexing matrix. This indexing matrix is then used in the two-ensemble simulation function, which outputs a $2N(n+b+1)$ vector containing the instantaneous frequencies and instantaneous changes in synaptic weights. This function can therefore be used to simulate Equation 2.20. We use these functions for calculations in Chapter 5.

This chapter established Python functions that can be used to model single-ensemble and two-ensemble systems of oscillators following Kuramoto–Sakaguchi dynamics under STDP with random coupling. The introduction of Numba JIT compilation makes these functions practical despite their reliance on nested loops. The development of these functions proceeded by replicating published results, from the classical Kuramoto

model, to the Kuramoto–Sakaguchi model with STDP under cyclic coupling. These replications provide numerical evidence that results obtained from these functions are trustworthy. In the next two chapters, these functions are used to obtain novel results. In Chapter 4, the single-ensemble function, together with the random adjacency-matrix helper function, is used to investigate the effect of random coupling on memory storage in Kuramoto–Sakaguchi ensembles. In Chapter 5, the two-ensemble function, together with the two-ensemble adjacency-matrix and indexing-matrix helper functions, is used to investigate whether two memory-encoding ensembles can be synchronised without permanently altering their memory characteristics.

Chapter 4

Effects of Random Coupling

This chapter is the first to present novel results. It addresses the objectives of extending the work of Ansariara et al. [104] and Botha et al. [69] by introducing a network with random coupling and investigating the effects of random coupling on the synchronisation behaviour of the network. The Kuramoto–Sakaguchi model with STDP introduced in Chapter 2 is simulated under random coupling using the functions described in Section 3.5.

In Section 4.1, we perform experiments that investigate the system with random coupling for small frustration. Then, in Sections 4.2 and 4.3, we reproduce the experiments performed by Botha et al. [69]. Section 4.2 investigates the behaviour of the model with random coupling when the frustration is in the chimera-state range observed by Botha et al. [69]. Finally, Section 4.3 investigates the behaviour of the model with increasing frustration.

4.1 Phase Synchronisation and the Emergence of a Pacemaker

This section investigates the behaviour of the Kuramoto–Sakaguchi model with STDP under the random network topology presented in Equation 2.9. This investigation uses the parameter choices employed by Ansariara et al. [104], and its findings are compared with those of Ansariara et al. [104] for cyclic connectivity. The experiments of this investigation are performed using the functions in Section 3.5.

Ansariara et al. [104] observed phase synchronisation with the emergence of a pacemaker for the parameters $N = 101$, $n = 50$, $\epsilon = 0.1$, $\tau_p = \tau_d = 0.05\pi$, $A = 1.0$, and $\alpha = 0.015\pi$. Here, we perform experiments with identical parameters under random connectivity. For our initial conditions, we set $\omega_i = 1.0$ to make the oscillators identical. The phases of the oscillators were chosen from a uniform distribution on $[-\pi, \pi]$. An experiment with these conditions was performed for a transient time of $T = 3000$ with a sampling time of $dt = 0.5$. The instantaneous oscillator phases and synaptic weights were recorded every 256 sampling steps. Furthermore, the running averages of the oscillator frequencies were calculated over 256 steps and recorded. After some time, the oscillator frequencies, phases, and synaptic weights reached a steady-state. The observed steady-state oscillator frequencies, synaptic connectivity pattern, and phases are presented in Figure 4.1.

Figure 4.1(a) shows that after reaching a steady-state, the running-average oscillator frequencies are identical and equal to the natural frequencies of the oscillators. In this case, there is phase-synchronised behaviour, i.e. the instantaneous frequencies are also identical, meaning that the relative phase differences of the oscillators remain constant, as can be seen by comparing Figure 4.1(c) and (d). For such a phase-synchronised solution, all phases advance at the same rate.

In Figure 4.1(c) and (d), oscillator $i = 70$, indicated by the blue triangle, has the leading phase, while oscillator $i = 87$, indicated by the green triangle, has the lagging phase. In the instantaneous synaptic strength matrix shown in Figure 4.1(b), the row corresponding to $i = 70$ is blank, indicating that the synaptic strengths from presynaptic oscillator 70 to its postsynaptic oscillators evolve to zero. Hence, the phase dynamics of this oscillator are not influenced by the other oscillators. In contrast, as discussed in Section 2.1, if $C_{ij} \rightarrow 0$, then $C_{ji} \rightarrow 1$. This is observed in column $j = 70$, where all available connections (all 50 of them) are equal to 1. This indicates that when oscillator 69 is the postsynaptic oscillator, the coupling strength is maximal. Hence, the 70th oscillator influences the phase dynamics of the oscillators it is connected to. These rows and columns are indicated in blue in Figure 4.1(b). In short, the 70th oscillator has the leading phase and influences other oscillators it is connected to without being influenced by any of them. Oscillators with this behaviour are called pacemakers [104].

For the 87th row, marked in green in Figure 4.1(b), all 50 initial couplings are equal to one, indicating that the 87th oscillator maintains maximal connections with its postsynaptic oscillators. In contrast, the 87th column, also marked in green, has coupling strengths that have evolved to zero, indicating that the 87th oscillator receives

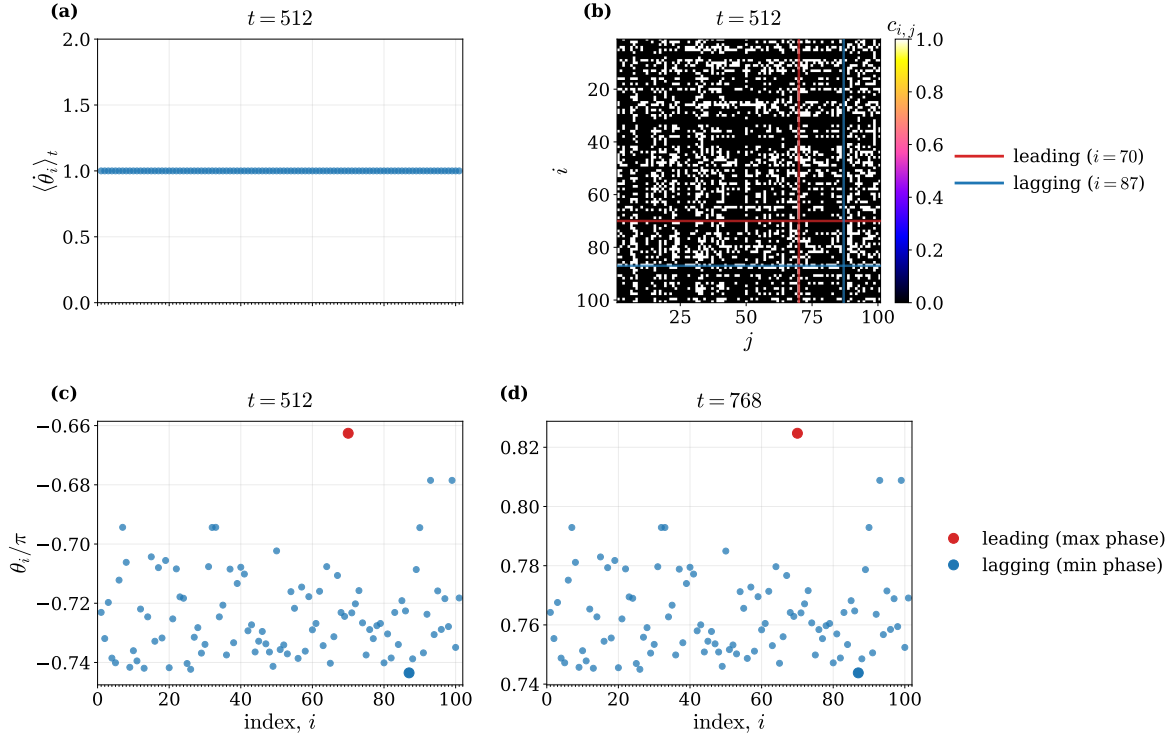


Figure 4.1: The steady-state behaviour of the Kuramoto–Sakaguchi model with STDP under random coupling for $N = 101$, $n = 50$, $\epsilon = 0.1$, $\tau_p = \tau_d = 0.05\pi$, $A = 1.0$, $\alpha = 0.015\pi$, and $\omega_i = 1$. (a) Running-average oscillator frequencies after reaching a steady-state. (b) Matrix of instantaneous synaptic strengths between presynaptic oscillator i and postsynaptic oscillator j after reaching a steady-state. The blue and green rows and columns are provided as eye guides indicating the rows and columns of the leading and lagging oscillators, respectively. (c) and (d) Instantaneous oscillator phases at $t = 512$ and $t = 768$.

no effective coupling from its presynaptic counterparts. Hence, it does not influence any oscillators that project to it. This makes it a lagging oscillator whose dynamics are influenced by other oscillators it is connected to, while it exerts little influence back on them.

The emergence of a pacemaker in Figure 4.1(c) and (d) is not surprising. Ansari-ara et al. [104] also observed the emergence of a pacemaker under identical conditions. Our findings show that introducing the random network does not change the emergence of a pacemaker or the qualitative nature of phase synchronisation in the system. Furthermore, the dynamics of the pacemaker and lagging oscillator suggest that the synaptic map formed by STDP depends on the ordering of oscillator phases: oscillators with leading phases retain more influence on oscillators they are connected to, while oscillators with lagging phases lose influence.

4.2 Effect of Random Coupling on Chimera States

In 2022, Botha et al. [69] showed that the Kuramoto–Sakaguchi model with STDP and cyclic coupling had a non-stationary chimera state for the parameters of $N = 60, n = 40, \tau_p = \tau_d = 0.001\pi, \epsilon = 0.01, A = 1, \alpha = 1.478$, and $\omega_i = 1$. They set up the initial conditions such that existing connections had the maximum coupling strength of $A = 1$, and the initial oscillator phases θ_i were given by $\theta_i = 6 r_i e^{-30x_i^2}$, where $r_i \sim \mathcal{U}(-0.5, 0.5)$ are independent random variables and $x_i = -0.5 + \frac{i}{m-1}$ for $i = 0, \dots, m-1$. This definition enabled some oscillators to have identical phases, while other oscillators had different phases. The system was evolved for a transient time of $T = 400000$ with a sampling time of $dt = 0.5$.

For the selected parameters, the system does not reach a steady state where the oscillator phases and frequencies and instantaneous couplings stop evolving. Here, we tracked the average synaptic strengths over time, and the results are shown in Figure 4.2. The figure shows that the average coupling tends to 0.5 as the system evolves, and when this average is reached, it is maintained. The average of 0.5 is approached because the individual pairwise connections assume a normal distribution with a peak at 0.5.

The instantaneous phases and synaptic strengths were recorded every 256 samples, and the frequencies were calculated as running averages in the same number of samples. The results of this experiment are shown in Fig. 4.3.

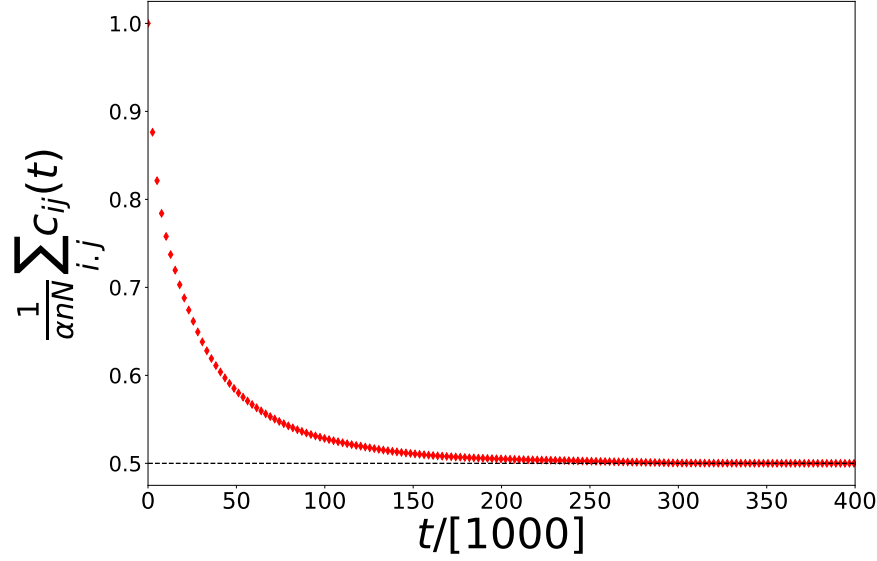


Figure 4.2: Time evolution of the average synaptic coupling for $N = 60, n = 40, \tau_p = \tau_d = 0.001\pi, \epsilon = 0.01, A = 1, \alpha = 1.478$, and $\omega_i = 1$.

Figure 4.3 (a) and (b) show the running average oscillator frequencies at $t = 256$ (a) and the corresponding instantaneous synaptic strengths. At this time point, the time-averaged frequencies mimic the distribution of the phases, and the coupling strengths are slowly varying away from the maximum strength of $A = 1$. The second panel, (c) and (d), shows the evolution of the system after $t = 31488$. The running average frequencies have differentiated into two groups, one that is coherent and one that is incoherent. The coherent group has the lowest frequency in the figure, forming a line of 5 oscillators. The other oscillators are in the incoherent group, with their frequencies randomly scattered above the frequency of the coherent oscillators. The instantaneous synaptic strengths have evolved such that the oscillators in the coherent group have the largest coupling strengths with their neighbours, both along the rows and columns of their indices, indicating that they both more strongly influence and are more strongly influenced by the oscillators they share synapses with compared to the oscillators in the incoherent group.

For the next panel, (e) and (f), $t = 156672$, and more oscillators are part of the coherent group. Further, the instantaneous synaptic strengths matrix shows that the coupling strengths of the oscillators that were initially in the coherent state are decreasing to the average coupling of the general system. (g) and (h) show the running average frequencies and instantaneous synaptic strengths at $t = 312576$. At this time,

the number of oscillators in the coherent group has increased to 19 oscillators. This number is more or less retained throughout the rest of the evolution of the system. The instantaneous synaptic strengths are more uniform, with no oscillators having outlying coupling strengths. Finally, (i) and (j) show the running average frequencies and instantaneous synaptic strength after the average synaptic strength stabilises. Here,

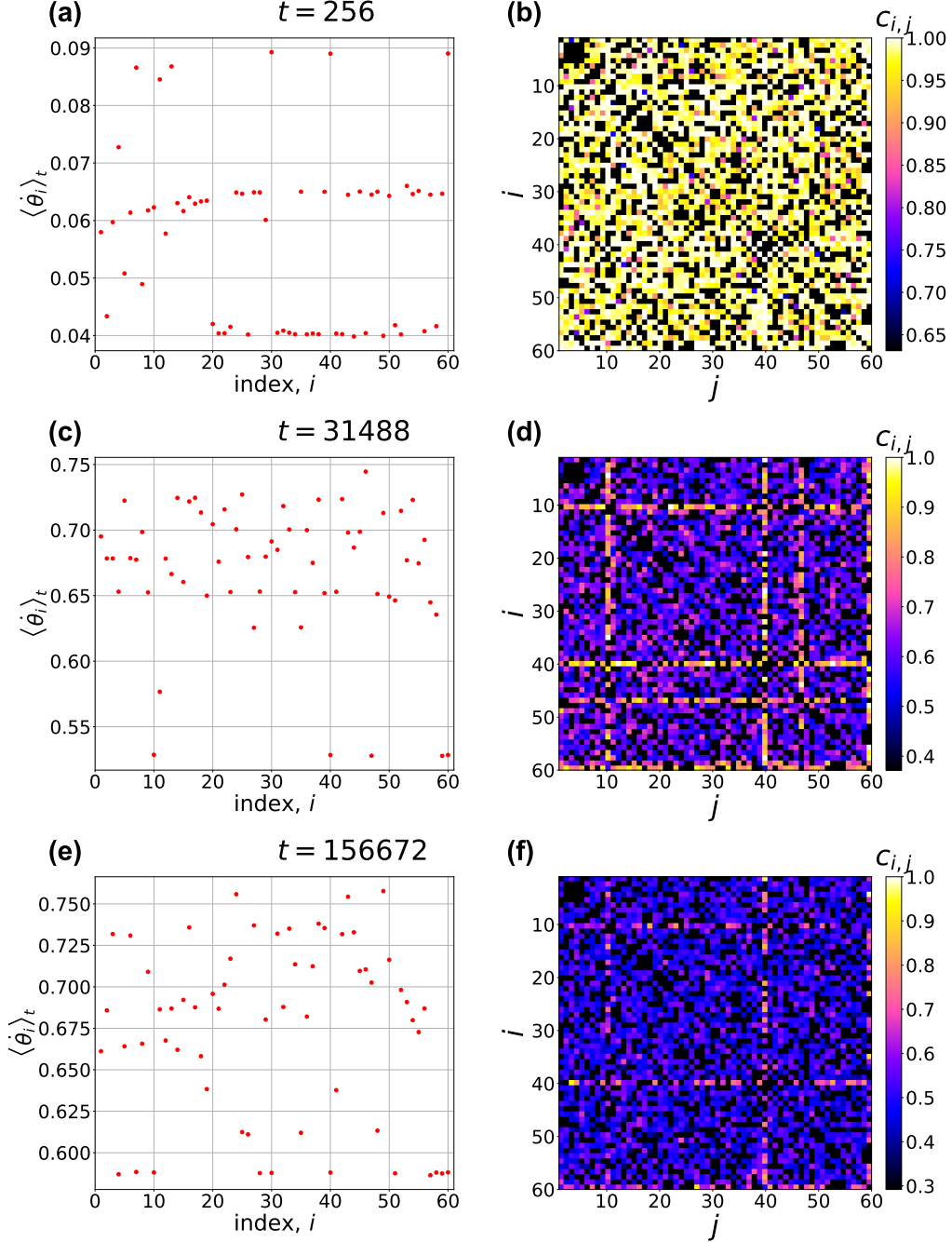


Figure 4.3: Continued on the next page.

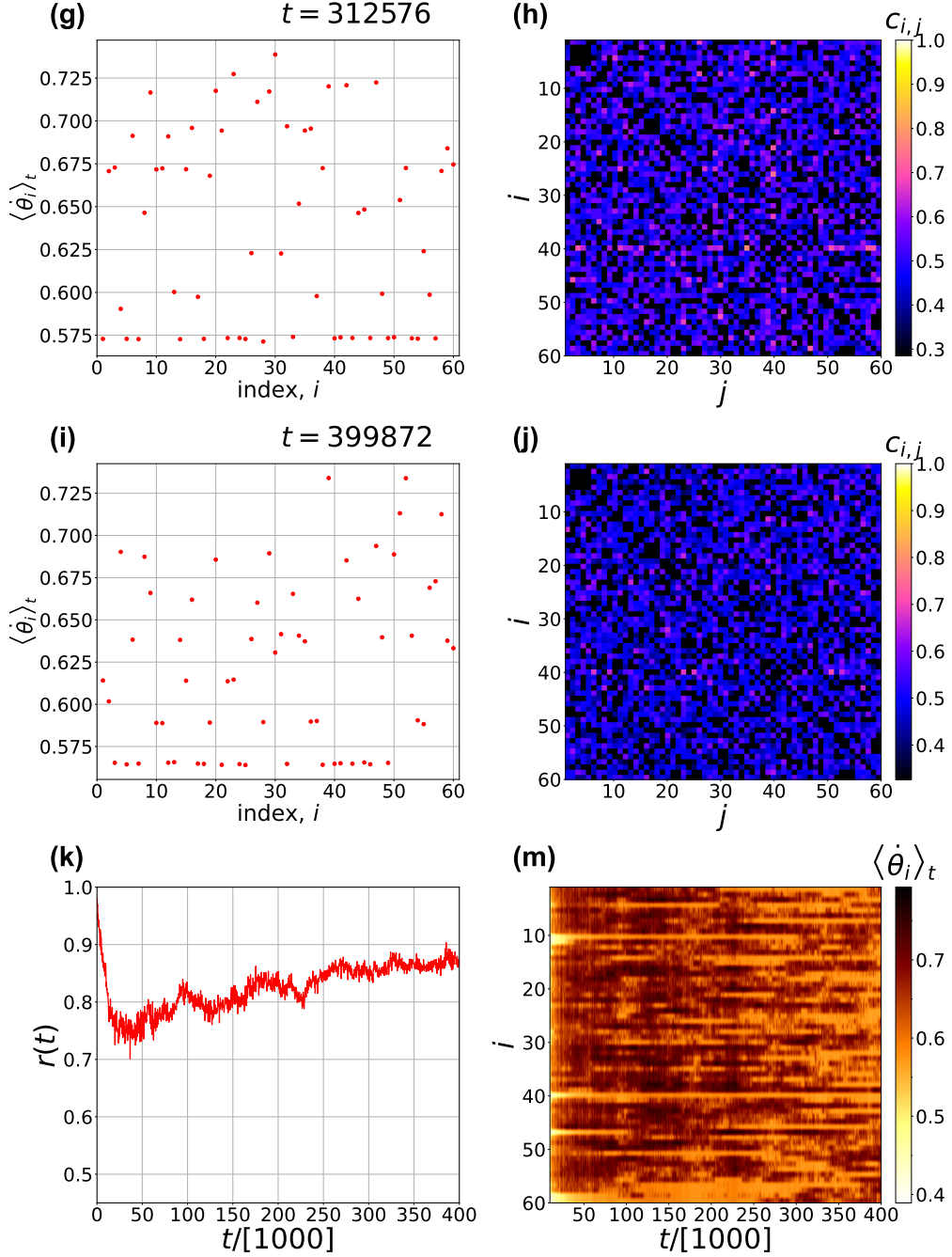


Figure 4.3: Time evolution of oscillator frequencies and coupling strengths for $N = 60, n = 40, \tau_p = \tau_d = 0.001\pi, \epsilon = 0.01, A = 1, \alpha = 1.478$, and $\omega_i = 1$. (a, c, e, g, i) The running-averaged frequencies of the oscillators at different time points. (b, d, f, h, j) Instantaneous coupling strength matrices at the times indicated in the corresponding running average frequency plot. (k) Time evolution of the order parameter for the system and that of the (m) time-averaged oscillator frequency for each oscillator.

$t = 399872$, and the structure of the oscillator frequencies remains the same as that in (e) and (h), with some oscillators becoming entrained by the coherent group and some becoming incoherent. The synaptic coupling does not have a defined structure, unlike in the earlier figures.

Figures (k) and (m) present the time-evolution of the order parameter and running average frequencies of the oscillators, respectively. Figure (k) shows that the system first has a high level of synchronization that decreases before increasing to oscillate about 0.85. In (m), the heat map indicates the value of the running average frequencies over time, and lower values are set to be brighter than higher values. The figure shows that some oscillators are in the coherent group—indicated by brighter columns—while others are outside this group. Oscillators can remain in this group indefinitely, while others can be entrained by the group at later times, and others are never entrained at all. Oscillators can enter and leave this group, but there is always a set of oscillators in the coherent group.

In the context of this study, the coherent group of oscillators can be considered to be neurons that fire in synchrony when a memory is learned. The motion of oscillators in and out of this group can be viewed as different neurons being activated to represent certain attributes of the learned content. However, it is difficult to state this definitively because the map of synaptic strengths, which is thought to store learned content, is always changing because the frequencies of the oscillators are never fixed.

In this experiment, we investigated the effect of introducing random coupling in the parameter regime where Botha et al. [69] observed chimera states. The results show that introducing random coupling changes the nature of the chimera state. We observe the coexistence of coherent oscillators with incoherent ones, with oscillators entering and leaving this group over time. The chimera-state form observed here does not assume the structure observed by Botha et al. [69]. The chimera state observed by Botha et al. [69] was such that the coherent group of oscillators was a group of nearest neighbour oscillators and so was the incoherent group. Further, the incoherent oscillators followed a downward-opening parabola. This chimera state moved across the oscillators over time [69].

We observe a different chimera state. Early in the evolution of the system, the entrained group is formed by oscillators that have large synaptic coupling strengths with the oscillators they are connected to. This group grows over time, with some oscillators joining and leaving at the steady-state. At present, it is unclear whether there is a pattern in the motion of the oscillators in and out of the coherent group. This result shows

that introducing the random network changes the nature of the model's chimera state. First, it eliminates the condition that coherent oscillators must be nearest neighbours and replaces the motion of the chimera with the motion of individual oscillators.

4.3 Effect of Random Coupling on the Interval of Stable Chimera States

This section investigates the effect of random coupling on the interval of stable chimera states observed by Botha et al. [69]. For the experiment, the parameters were set to $N = 40$, $n = 28$, $\tau_p = \tau_d = 0.001\pi$, $\epsilon = 0.01$, and $\omega_i = 1$. The frustration α was increased from 0 to 2π in steps of $\pi/500$. At each value of α , the system was allowed to evolve over a transient time of $T = 400\,000$ with a sampling time of $dt = 0.5$. The instantaneous phases and synaptic strengths were written to a file every 256 samples, and running-average frequencies were calculated over windows of 256 steps and written to a file. The next set of figures shows the results of these experiments.

Figure 4.4 shows the evolution of the order parameter and the maximum and minimum running-average frequencies as functions of the frustration parameter. The order parameter and running-average frequencies are those obtained at the end of the transient time for each experiment. The figure shows that the order parameter is equal to 1 for very small frustration and remains close to 1 thereafter. At the same time, the maximum and minimum running-average frequencies are equal and initially equal to the natural frequency of the oscillators ($\omega_i = 1$), after which they decrease as the frustration increases.

At approximately $\alpha = 1.231$, the order parameter begins to fluctuate before decreasing to values below 1 and oscillating around 0.85. The maximum and minimum running-average frequencies increase suddenly around this value of α and become distinct, indicating that the oscillators are no longer fully synchronised. At $\alpha = \pi/2$, the order parameter decreases to values close to 0, while the running-average frequencies increase rapidly and remain distinct.

The minimum and maximum running-average frequencies merge again in the interval $\alpha \in [2.834, 3.549]$, while the order parameter remains close to 0. This indicates a phase-synchronised steady-state, where the oscillators have identical frequencies but non-trivial phase differences. Beyond this range and up to $\alpha = 3\pi/2$, the maximum

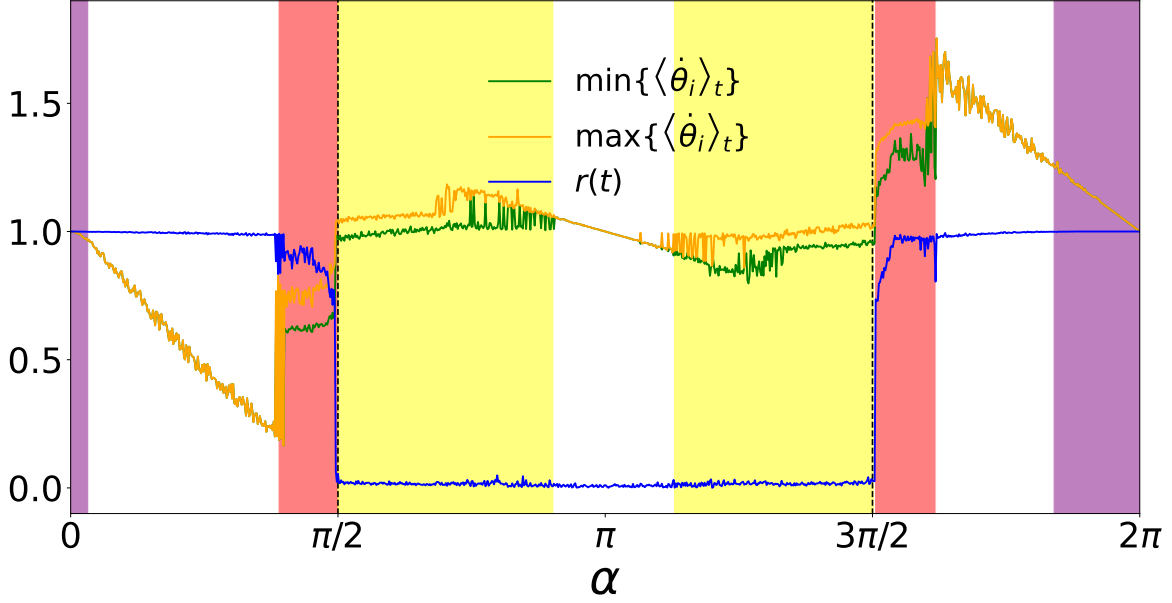


Figure 4.4: Behaviour of the system with $N = 40$, $n = 28$, $\tau_p = \tau_d = 0.001\pi$, $\epsilon = 0.01$, and $\omega_i = 1$ for different values of the frustration parameter α . The purple regions indicate intervals of α where there is full synchronisation, the white regions indicate intervals with phase-locked synchronisation, the pink regions represent intervals with the coexistence of coherent and incoherent groups, and the yellow regions indicate intervals with no synchronisation. The blue line shows the order parameter, while the orange and green lines represent the maximum and minimum running-average frequencies, respectively.

and minimum running-average frequencies diverge again, while the order parameter remains close to 0.

After $\alpha = 3\pi/2$, the order parameter increases rapidly but does not reach 1. The minimum and maximum running-average frequencies also increase and remain distinct. For $\alpha \geq 5.084$, the minimum and maximum running-average frequencies become equal again and decrease, while the order parameter approaches 1. Finally, at $\alpha = 5.780$, the order parameter reaches 1, and the minimum and maximum running-average frequencies return to the natural frequency of the oscillators as $\alpha \rightarrow 2\pi$.

The observations above suggest that the system exhibits distinct synchronisation properties for varying values of the frustration parameter. The purple regions indicate ranges of α where the oscillators are fully synchronised. That is, both the frequencies and phases of the oscillators are identical. This is indicated by the order parameter being equal to one and the minimum and maximum running-average frequencies being

equal.

The white regions show ranges where there is phase-locked synchronisation. In these regions, the oscillators have the same frequency, but their phases differ, with constant phase differences over time. The red regions indicate values of α where the system exhibits a chimera state, in which some oscillators are synchronised while others are not. The yellow regions represent intervals with no synchronisation. In these regions, the order parameter is close to zero, and the minimum and maximum running-average frequencies are unequal.

The subsequent figures show snapshots of the phases and frequencies for the different intervals of frustration. Figure 4.5 shows the running-average frequencies and instantaneous phases in the purple regions. Figure 4.5(a) shows the oscillator running-average frequencies at $\alpha = 0.00$, while Figure 4.5(b) shows the running-average frequencies and phases for $\alpha = 6.088$. In both panels, the running-average frequencies of the oscillators are identical, and the phases are also identical. This demonstrates full synchronisation in the purple regions.

Figure 4.6 shows the oscillator running-average frequencies and phases in the white regions. Figure 4.6(a) shows the running-average frequencies and phases at different times for $\alpha = 0.615$. Figures 4.6(b) and (c) show the running-average frequencies and phases at different times for $\alpha = 2.94$ and $\alpha = 5.208$, respectively.

For all panels, the running-average frequencies of the oscillators are identical, while the phases differ with constant phase differences, indicating phase-locked synchronisation. For panels (a) and (c), the spread of the oscillator phases is much smaller than that observed in (b). This follows from the order parameter behaviour in Figure 4.4. In the white regions on the left and right, represented by (a) and (c), the order parameter is close to 1, indicating that the phases are tightly clustered. In contrast, for the central white region, represented by (b), the order parameter is close to 0, indicating a wide spread of phases.

Figure 4.7 shows the running-average frequencies for the ranges indicated in red. In these ranges, the oscillators form coherent and incoherent groups. These include the frustration values studied in Section 4.2. Panel (a) shows frequencies for three frustrations in the first range, while panel (b) shows frequencies for three frustrations in the second range.

The figures in panel (a) show the running-average frequencies of the oscillators

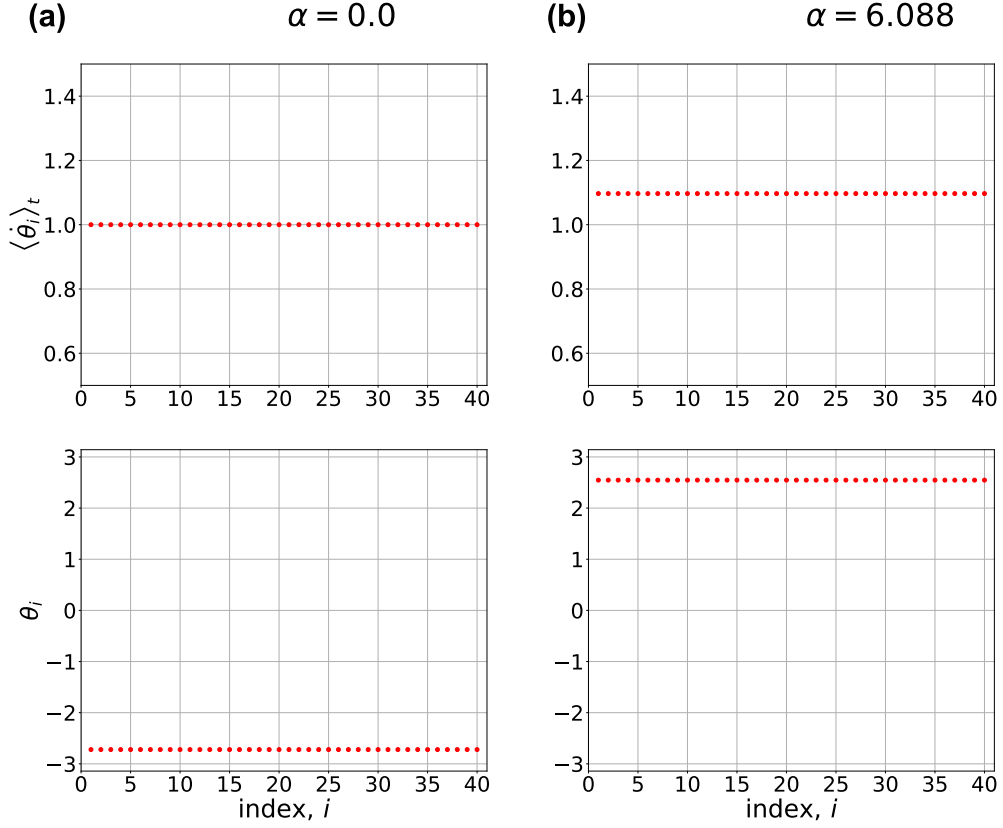


Figure 4.5: Example running-average frequencies and phases in the full synchronisation intervals for the system with $N = 40$, $n = 28$, $\tau_p = \tau_d = 0.001\pi$, $\epsilon = 0.01$, and $\omega_i = 1$. (a) and (b) Steady-state running-average frequencies and instantaneous phases for $\alpha = 0.0$ and $\alpha = 6.088$, respectively.

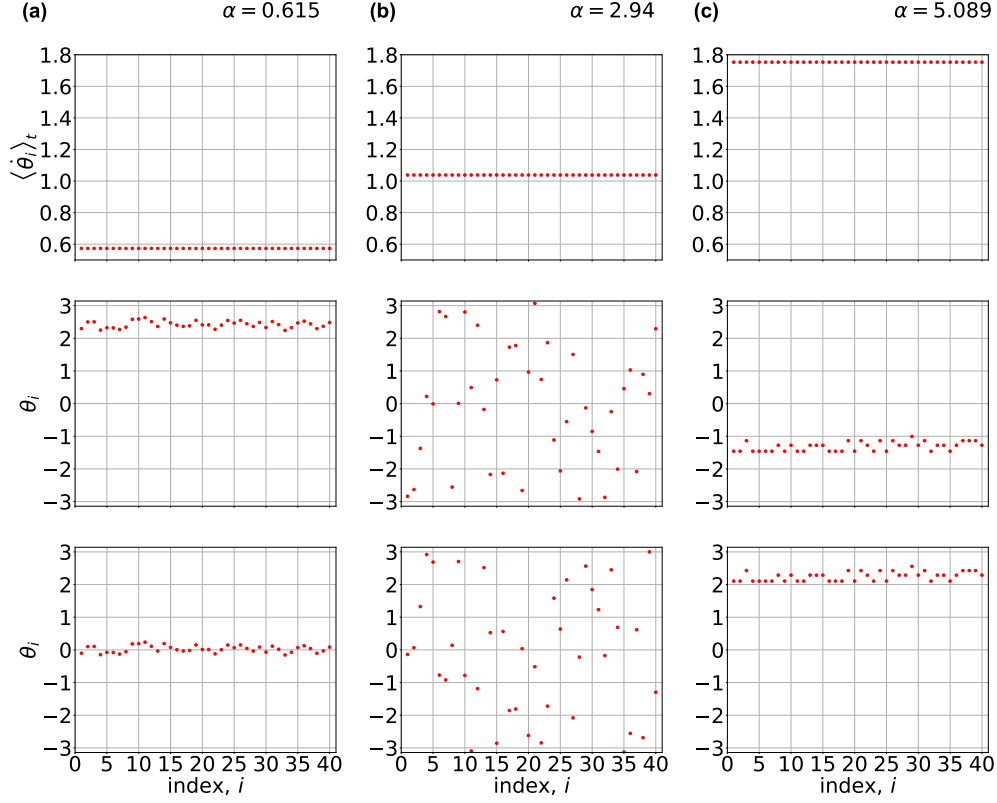


Figure 4.6: Example running-average frequencies and phases in the phase-locked synchronisation intervals for the system with $N = 40$, $n = 28$, $\tau_p = \tau_d = 0.001\pi$, $\epsilon = 0.01$, and $\omega_i = 1$. (a), (b), and (c) Steady-state running-average frequencies and instantaneous phases for $\alpha = 0.615$, $\alpha = 2.94$, and $\alpha = 5.089$, respectively.

for $\alpha = 1.275, 1.338$, and 1.476 . In these figures, the number of coherent oscillators decreases as the frustration increases. Furthermore, the frequency of the coherent group is lower than that of the oscillators in the incoherent group.

The figures in panel (b) show the running-average frequencies for $\alpha = 4.781, 4.844$, and 4.969 . In this panel, increasing α increases the number of oscillators in the coherent group. Additionally, the coherent group in these figures has a higher frequency than the oscillators in the incoherent group. The coherent groups in panel (b) also contain more oscillators than those in panel (a), which explains the larger order parameter observed in the second red region compared to the first.

Finally, Figure 4.8 shows the running-average frequencies for the yellow regions. In this figure, panel (a) shows the running-average frequencies for $\alpha = 2.192$ at different time points, while panel (b) shows the running-average frequencies for $\alpha = 4.09$ at the same time points. The running-average frequencies are irregular and vary across time. This lack of a consistent pattern indicates that the yellow region exhibits no form of synchronisation.

This section investigated the impact of random coupling on the interval of stable chimera states observed by Botha et al. [69]. The frustration parameter gives rise to different behaviours, including full synchronisation, phase-locked synchronisation, coexistence of coherent and incoherent oscillators, and no synchronisation.

Botha et al. [69] observed full synchronisation for $\alpha = 0$ to $\alpha \approx 1.370$ and $\alpha \approx 4.89$ to $\alpha = 6.283$. They also observed chimera states, where coherent oscillators coexist with incoherent oscillators, with the coherent group changing periodically over time, as described in Section 4.2. These states were observed for the intervals $\alpha \approx 1.370$ to $\alpha \approx 1.570$ and $\alpha \approx 4.71$ to $\alpha \approx 4.89$.

Further, they observe no synchronisation for short intervals just after $\alpha = 1.570$ and just before $\alpha = 4.71$. They observe phase synchronisation after the no-synchronisation regions, just after $\alpha = 1.570$ and just before $\alpha = 4.71$.

Our results show that introducing random coupling produces phase synchronisation for $\alpha \approx 0.100$ to $\alpha \approx 1.225$, $\alpha \approx 2.834$ to $\alpha \approx 3.549$, and $\alpha \approx 5.084$ to $\alpha \approx 5.780$, regions where Botha et al. [69] observed full synchronisation, where the oscillator phases and frequencies were identical. The ranges from $\alpha \approx 0.100$ to $\alpha \approx 1.225$ and from $\alpha \approx 5.084$ to $\alpha \approx 5.780$ were not observed by Botha et al. [69]. Further, the intervals where coherent oscillators coexist with incoherent oscillators expand to the wider ranges

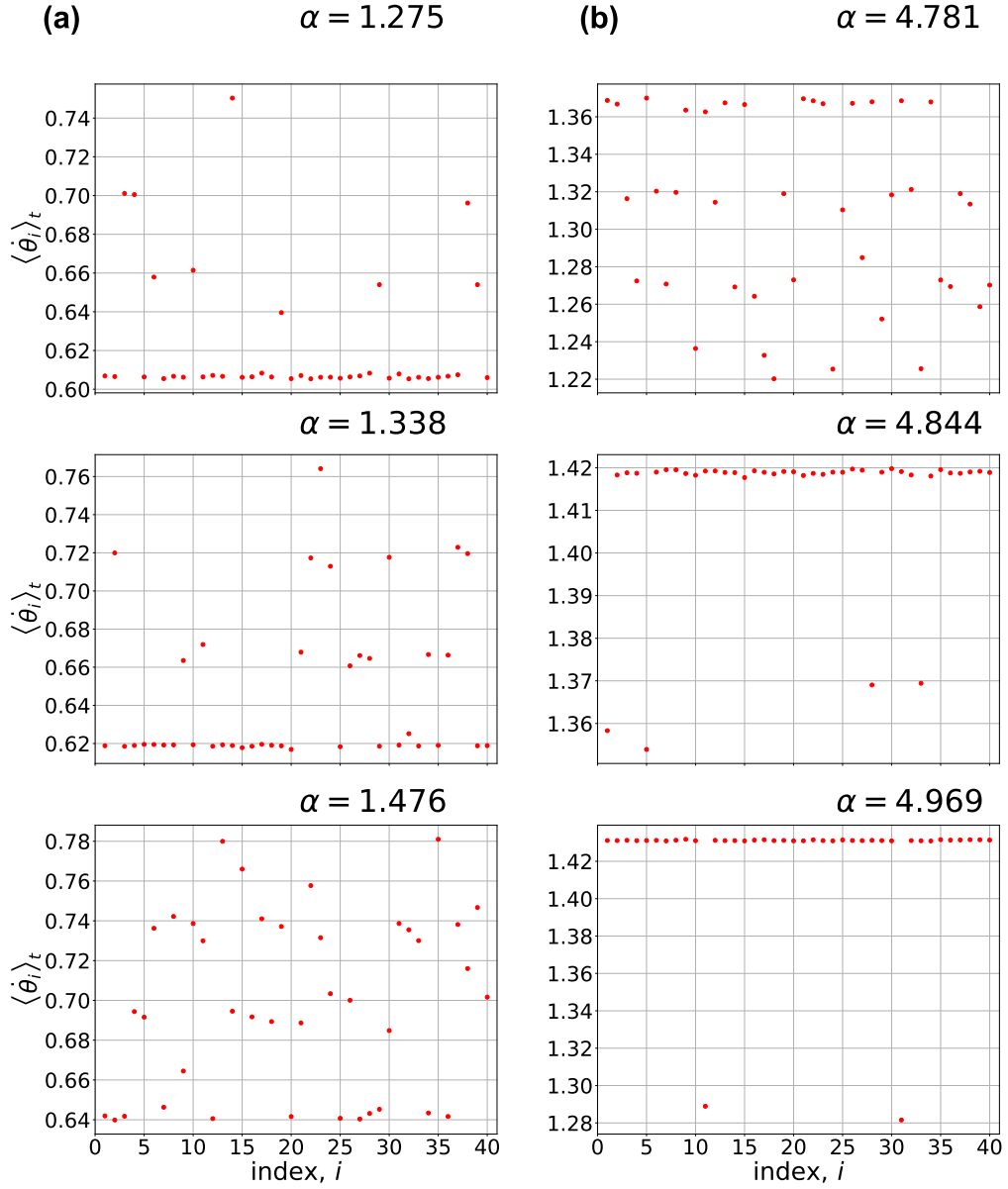


Figure 4.7: Example running-average frequencies in the interval where coherent and incoherent oscillators coexist for the system with $N = 40$, $n = 28$, $\tau_p = \tau_d = 0.001\pi$, $\epsilon = 0.01$, and $\omega_i = 1$. (a) and (b) Running-average frequencies for $\alpha = 1.275$, $\alpha = 1.338$, and $\alpha = 1.476$, and $\alpha = 4.781$, $\alpha = 4.844$, and $\alpha = 4.969$, respectively.

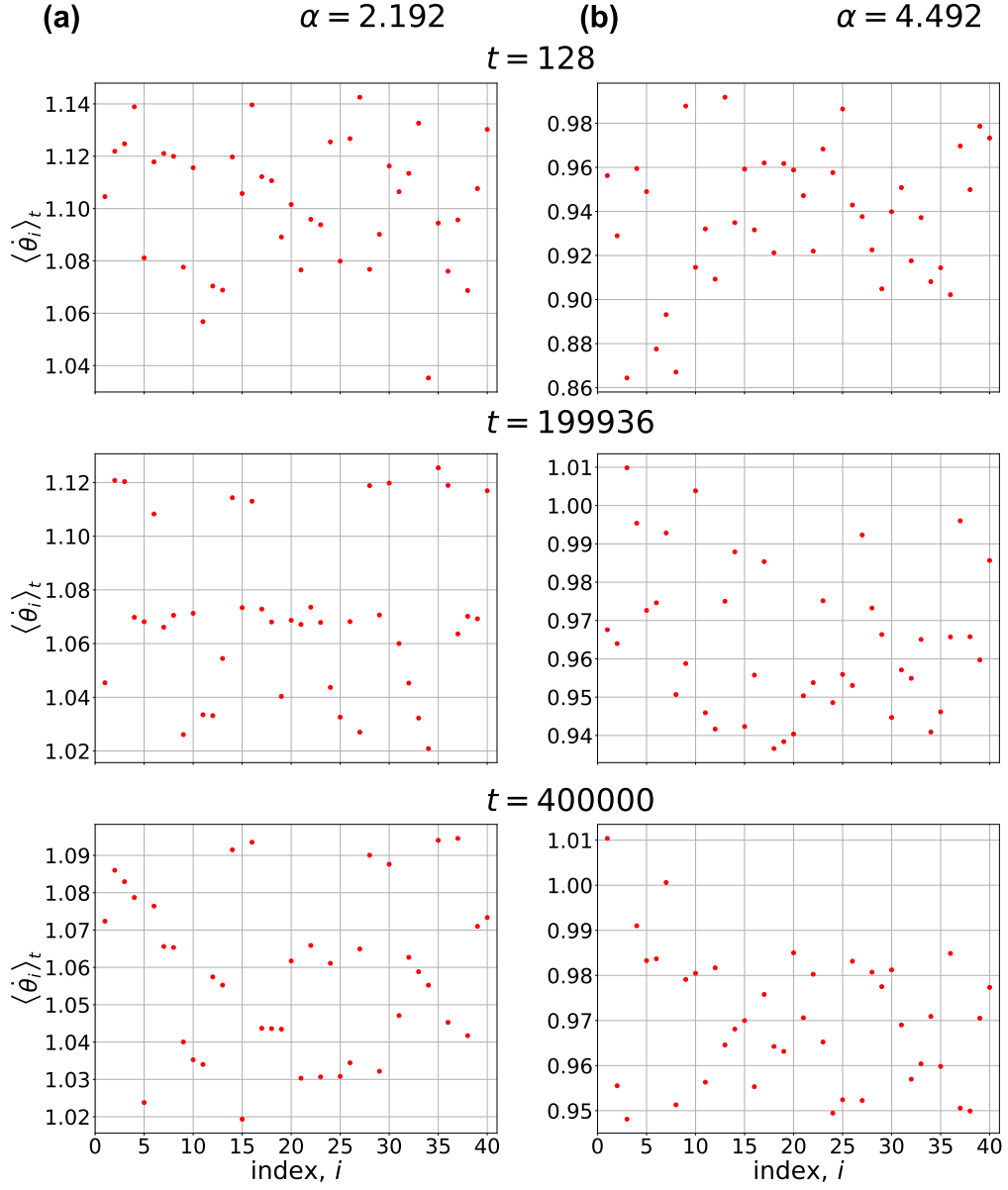


Figure 4.8: Example running-average frequencies in the interval with no synchronisation for the system with $N = 40$, $n = 28$, $\tau_p = \tau_d = 0.001\pi$, $\epsilon = 0.01$, and $\omega_i = 1$. (a) and (b) Running-average frequencies for $\alpha = 2.192$ and $\alpha = 4.969$, respectively.

of $\alpha \approx 1.225$ to $\alpha = 1.570$ and $\alpha \approx 4.731$ to $\alpha \approx 5.083$. In addition, random coupling introduces a wider range of no synchronisation, from $\alpha = 1.570$ to $\alpha \approx 2.834$ and $\alpha \approx 3.549$ to $\alpha \approx 4.717$.

These observations show that introducing random coupling reduces the interval of full synchronisation, while increasing the intervals of phase-locked synchronisation, the coexistence of coherent and incoherent oscillators, and no synchronisation.

If memories are encoded as the synaptic coupling of neurons along with the synchronisation of their firing rates and firing times, as suggested here, the extension of phase synchronisation suggests that more memories can be stored in the random coupling network than in the cyclic network. The number of possible phase-synchronised configurations is greater than that of full synchronisation, which is limited to a single fully synchronised state along with a limited number of synaptic maps under cyclic coupling. Further, the extension of the coexistence of coherent and incoherent oscillators has a similar effect. This range can be interpreted as a regime where memories in an ensemble are stored through the synchronisation of a subset of neurons in a neuronal population. Its extension provides a wider range of inter-neuronal signal delays that can support this type of memory.

In summary, introducing random coupling increases the ability to store memories under sparse coupling compared to cyclic coupling. Random coupling expands the intervals of phase-synchronised and chimera states. This expansion makes the model of neuronal ensembles with random coupling more biophysically plausible than those with cyclic coupling by accommodating the storage of more memories over a wider range of neuronal signal delays represented by the frustration parameter.

Chapter 5

Simulations of Coupled Ensembles

In this chapter, we present results of the mathematical model of mental synthesis presented in Section 2.6. These results are used to assess whether the model is suitable for modelling mental synthesis. Specifically, the chapter investigates whether two ensembles can be synchronised while retaining their memory properties.

As shown in Chapter 4, the Kuramoto–Sakaguchi model with STDP and random coupling exhibits three equilibrium states: full synchronisation, where the oscillator phases and frequencies are identical; phase synchronisation, where the oscillators have the same frequency and constant phase differences; and chimera states, where a subset of the oscillators is synchronised while the other oscillators remain desynchronised. The phase-synchronised and chimera states are characterised by non-trivial phase and synaptic weight patterns, which represent the memories encoded by the underlying ensemble.

This chapter examines whether two ensembles that evolve to phase synchronous states can be synchronised while preserving their phase patterns. Ensembles that evolve to phase synchronous states are selected because their fixed phase patterns can be compared at different time points. Further, the comparison is limited to the phase patterns because, as shown in Section 4.1, the steady-state coupling maps observed for these states are determined by the phase pattern.

Experiments for this chapter were performed in three phases. First, two ensembles were initialised and allowed to evolve via Equation 2.20 with no interensemble coupling. This allowed the two ensembles to evolve independently into phase synchronous states. Next, the inter-ensemble node degree m was set to a non-zero integer less than N ,

introducing sparse, random coupling between the ensembles. The system was then allowed to evolve to a new equilibrium state. Finally, the inter-ensemble coupling was removed and the system was allowed to relax to a final equilibrium. At each stage, the running average frequencies, phases, and synaptic weights of the oscillators were recorded. This protocol enables the assessment of whether synchronisation between ensembles can be achieved without erasing their original phase structure.

The Python functions and scripts presented in Section 3.6 were used to simulate the above experiments. Two ensembles with the parameters of $N = 40$, $n = 28$, $\alpha = 1$, $\omega_i = 1$, $\tau_p = \tau_d = 0.001\pi$, and $\epsilon = 0.01$ in the phase synchronised state intervals observed in Section 4.3 were initialised and then connected according to Equation 3.5. The experiments were performed in three stages. During the first stage, the two ensembles were initialised independently without any inter-ensemble coupling and allowed to evolve for $T = 400,000$ time units at sampling steps of $dt = 0.5$, and the running averages for the frequencies were computed over 256 consecutive samples before being stored in a file. The instantaneous phases and synaptic weights were also recorded every 256 steps. The initial coupling between the neurons of the two ensembles were set to $A = 1$, and the phases were drawn from a uniform random distribution in the range of $[-\pi, \pi]$. During the second stage, from $T = 400000$ to 800000 time units, the inter-ensemble coupling was enabled. These interensemble couplings had an A of 1, and each oscillator neuron in one ensemble was connected to $m = 20$ neurons in the other ensemble. These connections were bidirectional and a neuron in one ensembles was equally likely to be connected to any neuron in the other ensemble. During the third and final stage, from $T = 800000$ to 1200000 time units, the inter-ensemble coupling was removed.

For the phase-synchronous regimes, the frustration parameter was varied from $\alpha = 0.100$ to $\alpha = 1.225$ in increments of 0.056, from $\alpha = 2.840$ to $\alpha = 3.550$ in increments of 0.035, and from $\alpha = 5.084$ to $\alpha = 5.78$ in increments of 0.034.

The purpose of this three-stage strategy was to investigate the change in the phase and synaptic weight patterns of the two ensembles with and without inter-ensemble coupling. The first stage sets up the memory encoding ensembles discussed in Chapter 4. The second stage introduces inter-ensemble couplings, reflecting the connection of separate memory ensembles by the LPFC during mental synthesis. The third stage removes these inter-ensemble couplings, returning the ensembles to their uncoupled state. We investigate whether the steady-state phase patterns observed in the first stage are present in the second and third stages. The retention of these states would

represent the retention of memory during and after mental synthesis.

The remainder of the chapter is structured as follows: Section 5.1 introduces the metrics that will be used to assess synchrony between the ensembles and measure the correlation of the phase patterns of the ensembles in different inter-ensemble connectivity regimes. Next, Section 5.2 presents the synchronisation results of the ensembles, showing the change in the synchronisation metric when the ensembles are uncoupled, coupled, and decoupled. Finally, Section 5.3 shows the results of the correlation analysis of the phase patterns across the investigated coupling regimes.

5.1 Metrics Used

This chapter measures two quantities: the synchronisation of the ensembles and the correlation of their memory properties, namely, the steady-state phase patterns across different regimes of inter-ensemble connectivity. The metrics used to quantify these properties are presented below.

5.1.1 Global Synchronisation Metric

In previous chapters, the degree of synchronisation within a single ensemble was quantified using the Kuramoto order parameter. To quantify the synchronisation between two ensembles, we calculated the order parameter over the entire $2N$ network:

$$r_{global}e^{i\Psi} = \frac{1}{2N} \sum_{i=1}^{2N} e^{i\theta_i}, \quad (5.1)$$

where $i \in \{1, \dots, 2N\}$, with oscillators $1 \leq i \leq N$ belonging to the first ensemble and $N + 1 \leq i \leq 2N$ belonging to the second ensemble.

r_{global} measures the degree of global synchronisation across the two ensembles. When $r_{global} \approx 1$, oscillators in both ensembles are synchronised, while $r_{global} \approx 0$ indicates no synchrony between the oscillators in the ensembles. The parameter can be close to zero for a phase synchronous state where the phases are evenly distributed in a unit circle, cancelling each other out in the complex average. The global order parameter is used in Section 5.2 to assess the synchronisation of two ensembles.

5.1.2 Phase Pattern Correlation Metric

To quantify memory preservation, we define a correlation metric for the phase patterns of the two ensembles. This metric compares patterns within the same ensemble at different time points.

During phase-locked synchronisation, the phase differences between oscillators remain constant. Hence, they can be used to characterise a phase synchronised state. The pairwise phase differences are defined as

$$\Delta_{ij} = \theta_j - \theta_i, \quad (5.2)$$

for all $i, j \in \{1, \dots, N\}$ in the first ensemble and $i, j \in \{N + 1, \dots, 2N\}$ in the second ensemble.

These differences form an $N \times N$ matrix Δ that is antisymmetric and has a zero diagonal. Because of the antisymmetry of the matrix, it is sufficient to consider only the upper triangular entries of the matrix. As shown in Figure 4.6, the phase locked state in the ranges (0.1, 1.225) and (5.084, 5.780) is in a narrow range, making the oscillator phase differences small. Using these phase differences in correlation measures can cause the measures to saturate to one even when there are small but meaningful changes in the underlying phase patterns. To address this limitation, the phase differences were standardised using z-scoring. For a vectorised set of the upper-triangular entries of the phase difference matrix $\mathbf{d}$, the normalised values are computed as

$$\tilde{d}_k = \frac{d_k - \mu_d}{\sigma_d}, \quad (5.3)$$

where k is the phase difference index and μ_d and σ_d are the mean and standard deviation of $\{d\}$, respectively.

The similarity between two standardised phase patterns $\tilde{\mathbf{d}}^{(1)}$ and $\tilde{\mathbf{d}}^{(2)}$ is then quantified using the Pearson correlation coefficient:

$$\rho = \frac{\sum_k \left(\left(\tilde{d}_k^{(1)} - \mu_1 \right) \left(\tilde{d}_k^{(2)} - \mu_2 \right) \right)}{\sigma_1 \sigma_2}, \quad (5.4)$$

where μ_1, μ_2 and σ_1, σ_2 are the means and standard deviations of the two vectors, respectively. The coefficient $\rho \in [-1, 1]$, where 1 indicates perfect agreement, 0 indicates no linear correlation, and -1 indicates perfect inverse correlation. This metric is used in Sections 5.3 to quantify the similarity of phase patterns under different inter-ensemble coupling conditions.

5.2 Synchronisation of Two Ensembles

This section investigates the synchronisation of two ensembles that are coupled after independently evolving to phase-synchronised states. The global order parameter introduced in Section 5.1.1 is used to quantify inter-ensemble synchronisation.

As noted above, experiments in the phase-synchronisation regime were conducted for the frustration intervals of $0.100 \leq \alpha \leq 1.225$ (step size 0.056), $2.840 \leq \alpha \leq 3.550$ (step size 0.035), and $5.084 \leq \alpha \leq 5.780$ (step size 0.034). Figure 5.1 shows the global order parameter for the *uncoupled* regime, referring to the system when there is no inter-ensemble coupling initially, *coupled* regime, referring to the system when the inter-ensemble coupling has been turned on, and *decoupled* regime, referring to the system when the inter-ensemble coupling has been turned off again. The inter-ensemble node degree was set to $m = 20$.

In Figures 5.1(a) and (g), the global order parameter oscillates with α at values less than one, indicating that the ensembles are not synchronised when uncoupled. In Figure 5.1(d), the global order parameter is close to zero because the individual ensembles are internally phase synchronised with a wide phase-locked range. Importantly, the global order parameter does not remain flat along the origin; instead, it varies with α , indicating that cross-ensemble synchronisation is not realised in the range $2.840 \leq \alpha \leq 3.550$ when the ensembles are uncoupled.

When inter-ensemble coupling is introduced (Figures 5.1(b) and (h)), the global order parameter is flat close to one across all values of α , indicating strong synchronisation between the ensembles for the frustration intervals of $0.100 \leq \alpha \leq 1.225$ and $5.084 \leq \alpha \leq 5.780$. In Figure 5.1(e), plotted for the system with interensemble coupling, the global order parameter is flat and close to zero across the entire range of $2.840 \leq \alpha \leq 3.550$. Despite this, the oscillators are phase locked across the ensembles. The low order parameter in this frustration range is due to the broad distribution of the oscillator phases, which leads to cancellation in the complex average during the computation of the order parameter.

After inter-ensemble coupling is removed (Figures 5.1(c), (f), and (i)), the global order is no longer flat close to unity for (c) and (i) and zero for (f). This change indicates that the synchronisation between the ensembles is lost. These results demonstrate that inter-ensemble coupling can induce synchronisation across ensembles and this synchronisation is not generally maintained after decoupling.

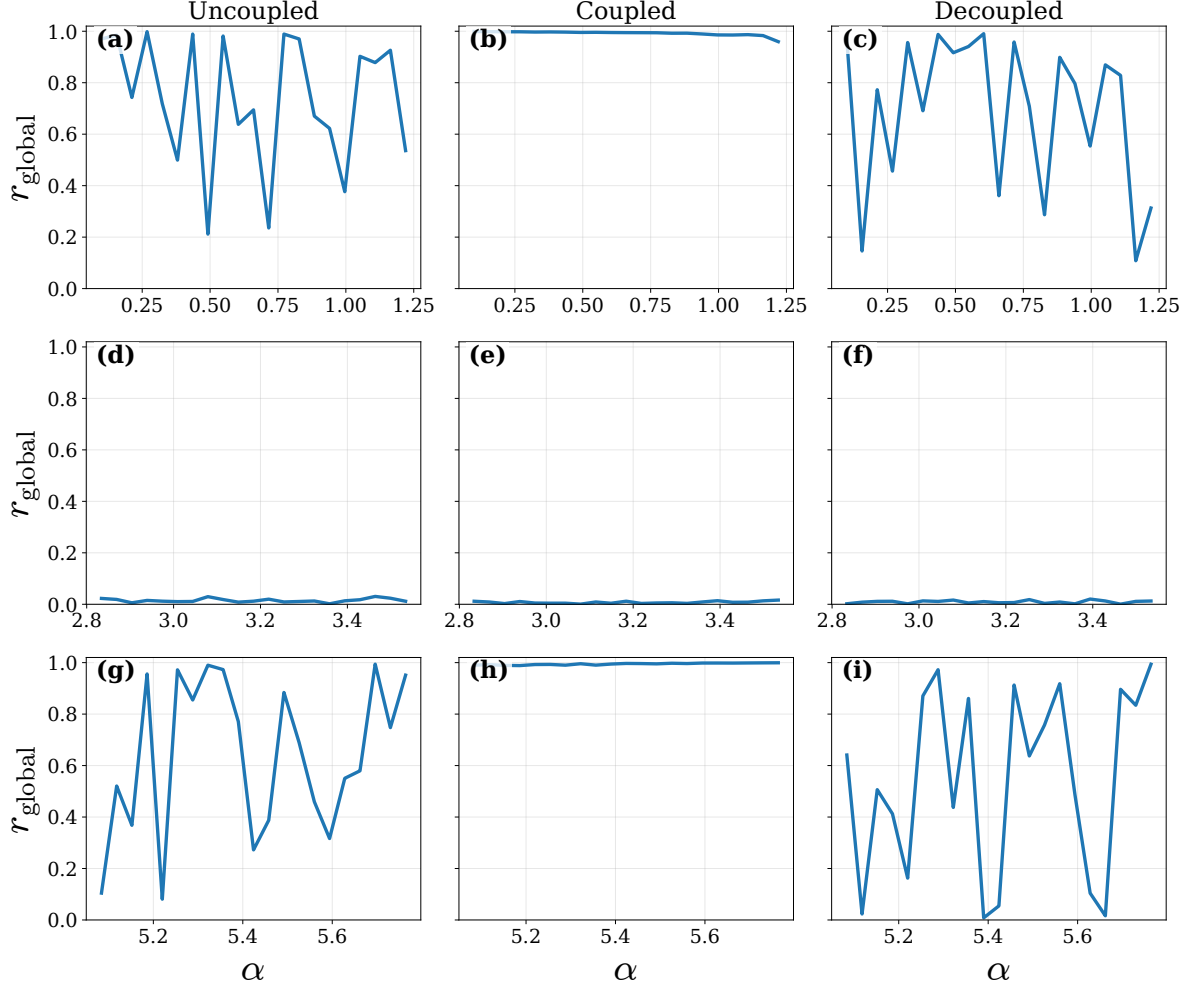


Figure 5.1: Global order parameter for the two ensemble system under the three phase-synchronous frustration intervals. (a - c) Correspond to the frustration range (0.1, 1.225), (d - e) (2.834, 3.549), and (g - i) (5.084, 5.780) with $b = 20$. (a, d, g) Ensembles uncoupled, (b, e, h) coupled, and (c, f, i) decoupled.

The difference of the average frequencies of the two ensembles provides further evidence of synchronisation. For Figure 5.2, we calculated the average frequency of each ensemble and then plotted the absolute difference of these averages with frustration. When the ensembles are coupled (Figures 5.2(b), (e), and (h)), this difference decreases by several orders of magnitude compared to the uncoupled and decoupled cases, indicating that the ensembles share a common frequency and are therefore synchronised.

Figures 5.1 and 5.2 provide summary figures of the synchronisation of the two ensembles when they are connected. Hence, Figures 5.3 and 5.4 show the running average frequencies and phases of individual oscillators in the two ensembles. The frequencies

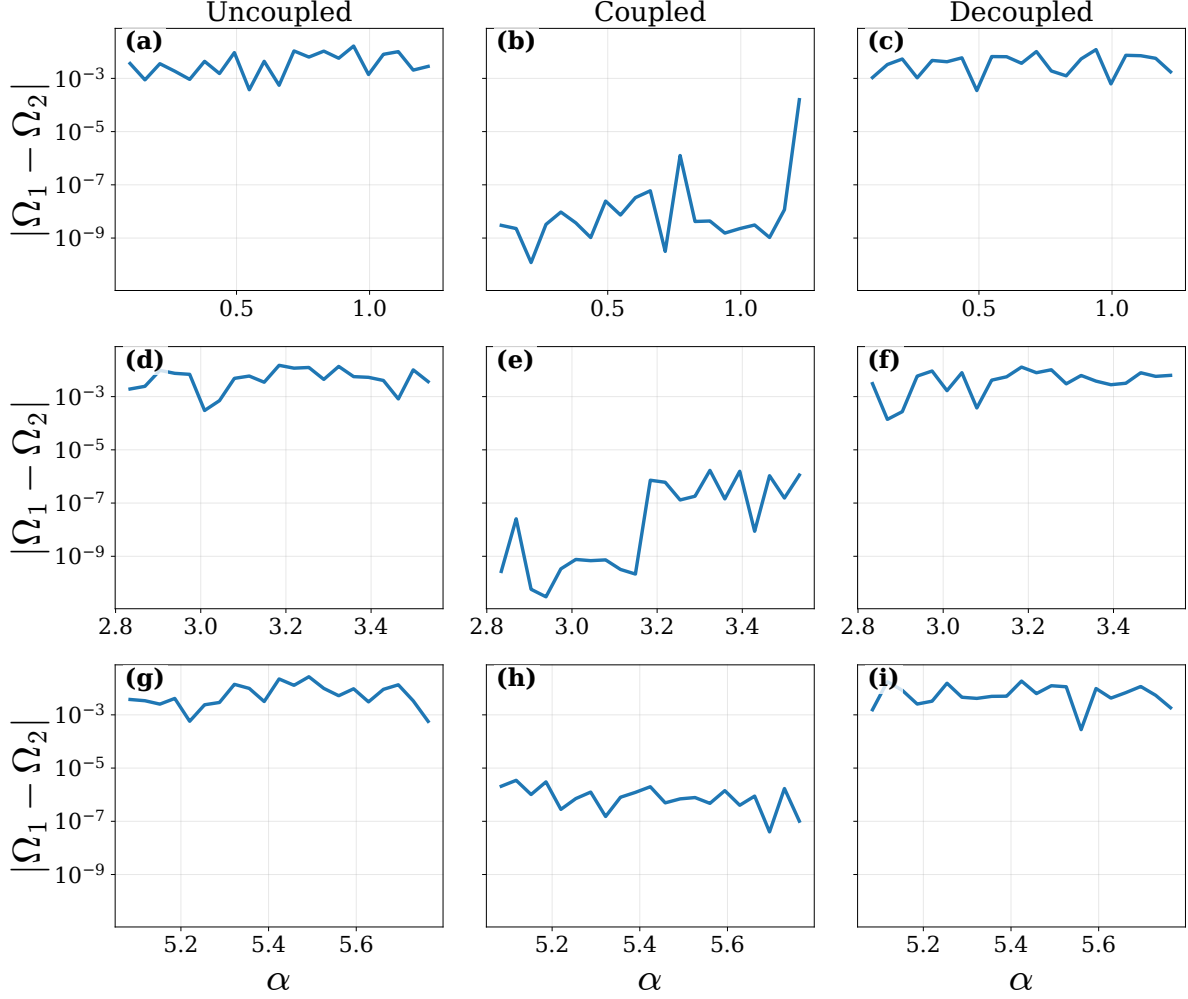


Figure 5.2: Difference of the average oscillator frequencies of ensembles 1 and 2 for the three phase-synchronous frustration intervals. (a - c) Correspond to the frustration range (0.1, 1.225), (d - e) (2.834, 3.549), and (g - i) (5.084, 5.780) with $m = 20$. (a, d, g) Ensembles uncoupled, (b, e, h) coupled, and (c, f, i) decoupled.

and phases are visualised for representative values of α for each frustration interval. In the figures, (a, d, g), (b, e, h), and (c, f, i) show the snapshot of the running average frequencies and phases, respectively, at the end of the period when ensembles are uncoupled, coupled, and decoupled, respectively. The top row corresponds to $\alpha = 0.604$, the middle row to $\alpha = 2.939$, and the bottom row to $\alpha = 5.084$.

In the uncoupled regime (panels (a, d, g)) in Figure 5.3, oscillators within each ensemble share a common running average frequency. However, the two ensembles have distinct frequencies, indicating a lack of inter-ensemble synchronisation. When coupling is introduced (panels (b, e, h)), all the oscillators assume the same frequency,

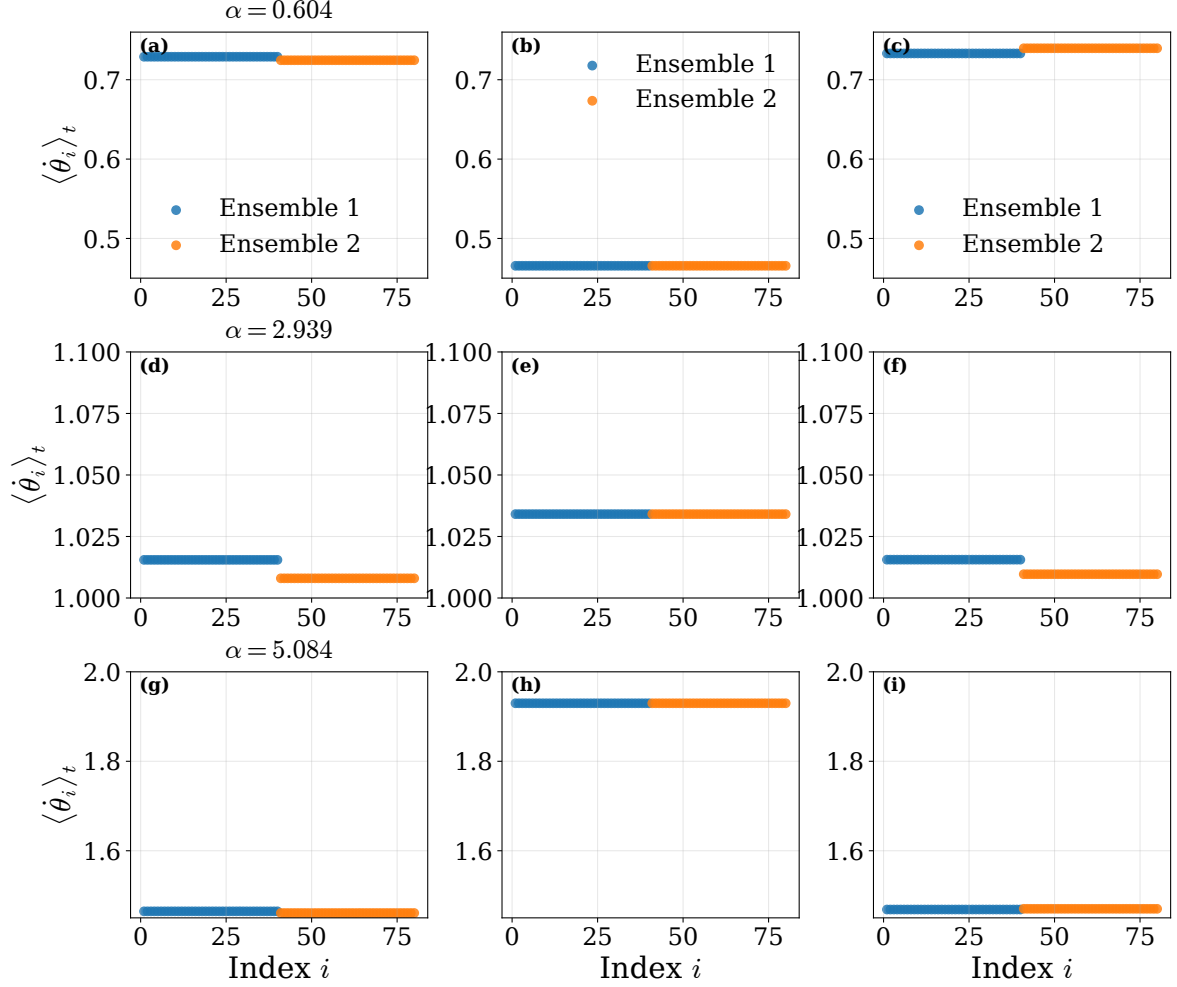


Figure 5.3: Running average frequencies of individual oscillators in the two ensembles. Blue dots denote oscillators in the first ensemble, and red dots denote oscillators in the second ensemble. Columns (a), (b), and (c) correspond to uncoupled, coupled, and decoupled conditions, respectively. Rows correspond to $\alpha = 0.604$, 2.939 , and 5.084 , respectively.

demonstrating synchronisation across both ensembles. After decoupling (panels (c, f, i)), the ensembles diverge to distinct frequencies, indicating the loss of synchronisation.

Figure 5.4 demonstrates similar results for the instantaneous phases. In the uncoupled regime (panels (a, d, g)), oscillators are phase-locked within each ensemble but not across ensembles. When coupling is introduced (panels (b, e, h)), phase-locking extends across both ensembles, indicating global phase synchronisation. After decoupling (panels (c, f, i)), this inter-ensemble phase-locking is lost, and each ensemble returns to an intra-ensemble phase-locked configuration.

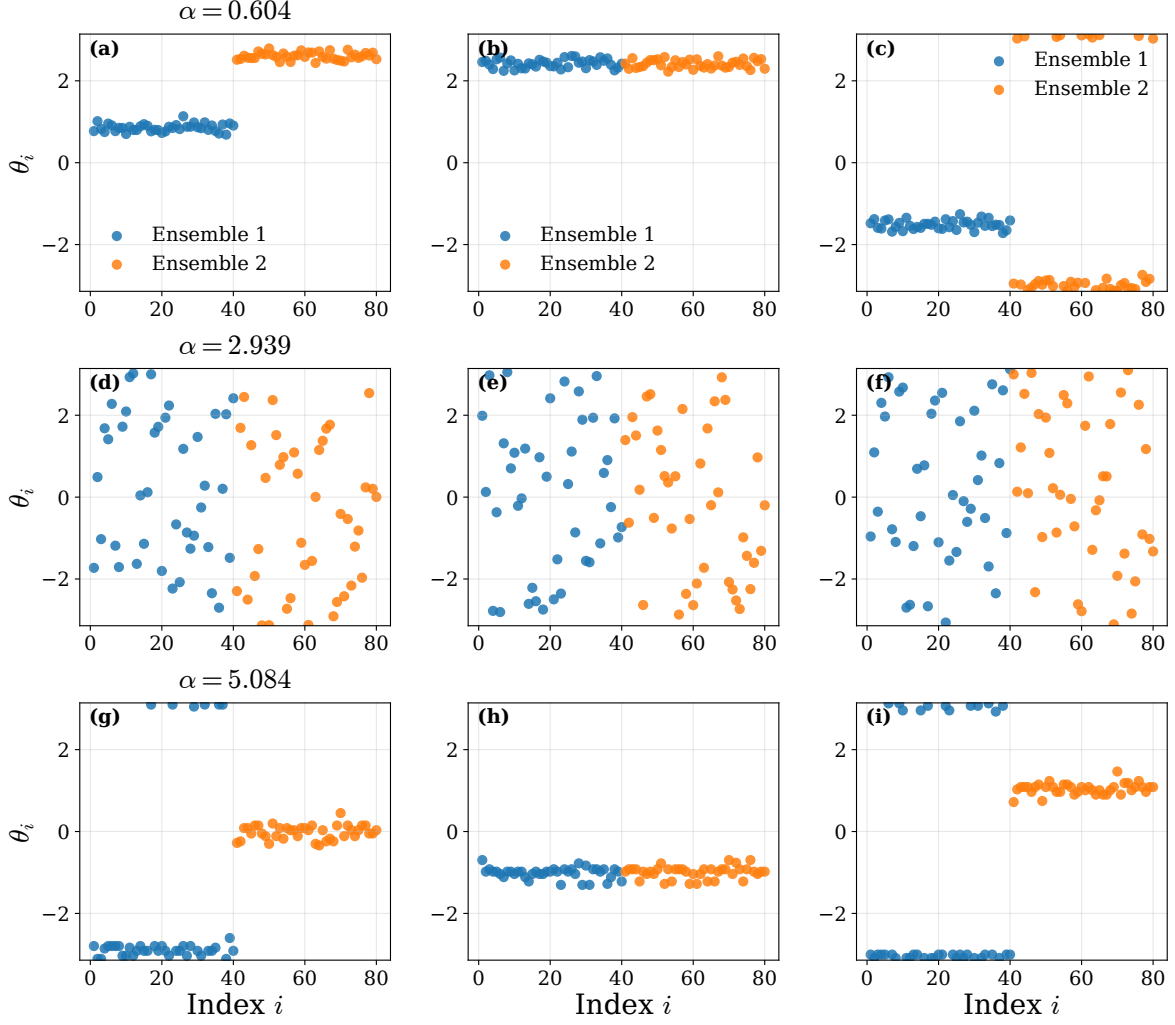


Figure 5.4: Instantaneous phases of oscillators in the two ensembles. Blue dots denote oscillators in the first ensemble and red dots denote oscillators in the second ensemble. Columns (a), (b), and (c) correspond to uncoupled, coupled, and decoupled conditions, respectively. Rows correspond to $\alpha = 0.604$, 2.939 , and 5.084 , respectively.

The results in this section demonstrate that inter-ensemble coupling enables synchronisation between two ensembles. This synchronisation is characterised by a global order parameter close to unity for narrow range synchronisation and zero for wide range synchronisation. The synchronisation is also demonstrated by the convergence of oscillator frequencies across ensembles. Importantly, the nature of synchronisation remains phase-locked, consistent with the intrinsic dynamics of the individual ensembles.

Having established that inter-ensemble coupling can synchronise two ensembles, the next question is whether this process preserves their memory properties. Specifically,

we investigate whether the phase patterns observed in the uncoupled state are retained during and after coupling. This question is addressed in the following section.

5.3 Retention of Memory Characteristics

In the previous section, we demonstrated that two ensembles can be brought into synchrony through inter-ensemble coupling. Achieving such synchronisation is essential for modelling mental synthesis, which has been hypothesised to occur when neuronal ensembles representing distinct mental objects become synchronised. However, synchronisation alone is insufficient. For the model to be useful in describing mental synthesis, the synchronised state must retain the memory characteristics of the individual ensembles.

In this section, we investigate whether the phase patterns that characterise a phase-synchronised ensemble are preserved when two ensembles are coupled and allowed to synchronise. To quantify pattern retention, we compute the Pearson correlation between the standardised phase-difference patterns of the same ensemble across different inter-ensemble coupling regimes.

During phase synchronisation, oscillators converge to a steady state in which phase differences remain constant while the phases drift at a common frequency. As shown in Figure 4.1, this phase-locked configuration is encoded in the synaptic weight structure through STDP. In the present framework, this steady phase pattern is analogous to the firing pattern associated with a particular memory. Consequently, if two memory-encoding ensembles are synchronised during mental synthesis, aspects of their original phase patterns should persist.

To assess retention, we compute Pearson correlations of the z-scored phase-difference patterns for three comparisons: (i) uncoupled versus coupled states, (ii) coupled versus decoupled states, and (iii) uncoupled versus decoupled states. These comparisons determine whether features of the independently formed phase patterns are preserved during coupling, whether features acquired during coupling persist after disconnection, and whether the original phase patterns are reinstated after decoupling.

These comparisons provide insight into different aspects of memory retention during mental synthesis. The comparison between the uncoupled and coupled states quantifies the degree to which features of the original phase patterns persist during inter-

ensemble synchronisation. In the context of mental synthesis, this retention represents memory traces that contribute to the construction of the synthesised mental image.

The comparison between the uncoupled and decoupled states measures the extent to which the original memory patterns are restored after the ensembles are disconnected. High correlation in this comparison indicates that the process of synchronisation does not permanently alter the stored memories but temporarily combines them in a reversible manner.

The results for the phase-synchronised frustration ranges are shown in Figure 5.5. For $\alpha \in (0.25, 1.125)$, the correlation between uncoupled and coupled phase patterns is approximately 0.5, indicating partial retention of the original phase configuration during synchronisation. A similar behaviour is observed for $\alpha \in (5.084, 5.780)$. In contrast, for $\alpha \in (2.834, 3.549)$, the correlation is approximately zero, indicating that synchronisation in this range is accompanied by a complete restructuring of the phase pattern.

For the comparison of the coupled and decoupled states, the correlation decreases from approximately 0.85 to 0.5 with increasing α in the interval $(0.100, 1.225)$, indicating that a substantial portion of the synchronised phase pattern persists after decoupling. For $\alpha \in (2.834, 3.549)$, the correlation remains near zero, showing that phase structures formed during coupling are not retained once the ensembles are disconnected. In contrast, for $\alpha \in (5.084, 5.780)$, the correlation remains near 0.5, indicating moderate retention of the synchronised configuration.

Finally, comparing the uncoupled and decoupled states reveals that, for $\alpha \in (0.100, 1.225)$, the correlation increases to a peak of approximately 0.85 near $\alpha \approx 1$ before decreasing again, indicating substantial restoration of the original phase pattern after disconnection. In the interval $(2.834, 3.549)$, correlations fluctuate around zero, suggesting that the original memory pattern is not reinstated. For $\alpha \in (5.084, 5.780)$, the correlation averages approximately 0.75, indicating strong recovery of the initial configuration.

These results demonstrate that, although inter-ensemble synchronisation occurs in all three phase-synchronised regimes, retention of memory characteristics depends on the value of the frustration. In particular, the interval $\alpha \in (2.834, 3.549)$ supports synchronisation but not memory retention. In contrast, the range $\alpha \in (5.084, 5.780)$ consistently supports both synchronisation and substantial pattern retention. The interval $\alpha \in (0.100, 1.225)$ contains parameter values, such as $\alpha \approx 1$, that simultaneously

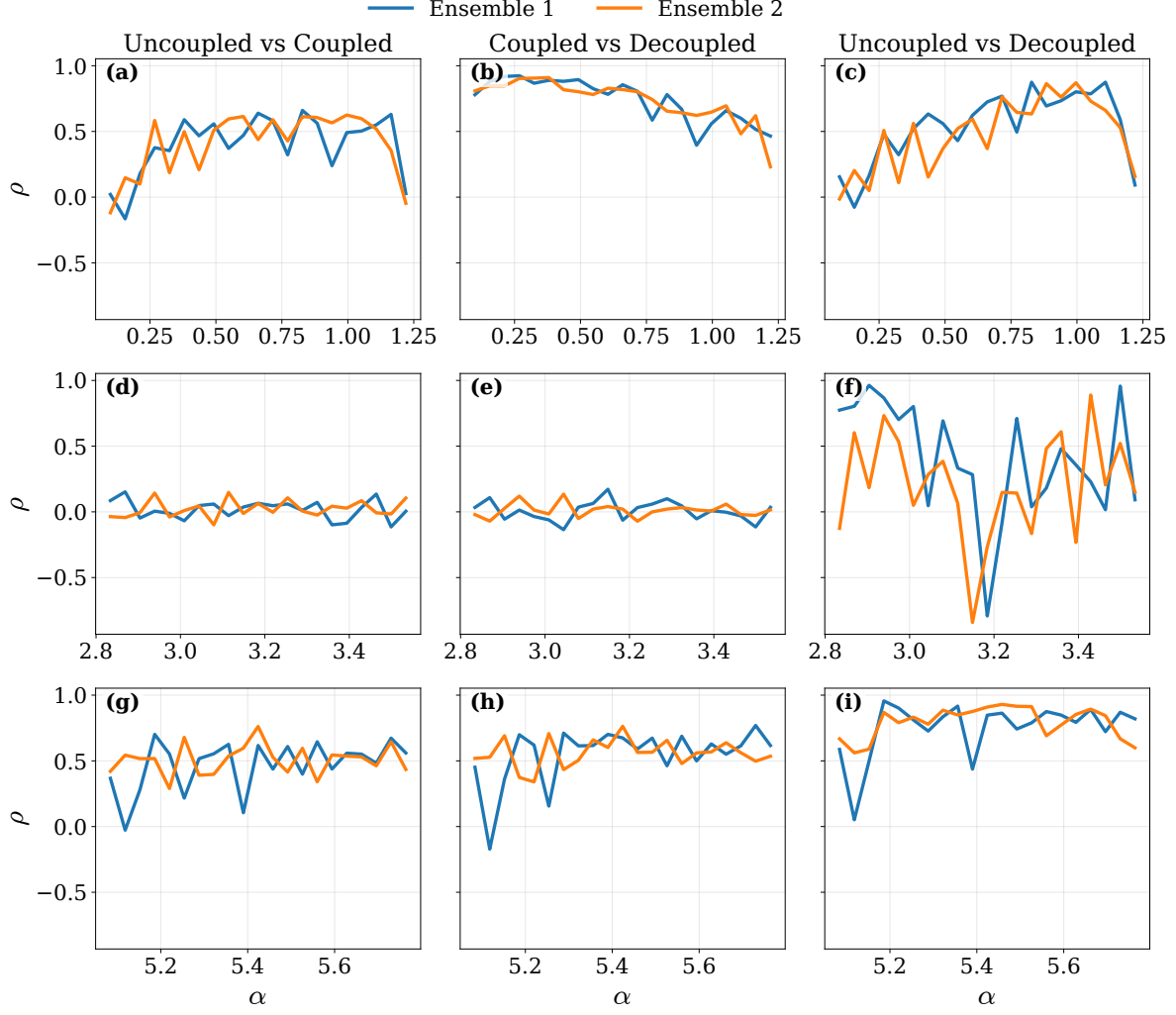


Figure 5.5: Pearson correlation on the z-score phase differences of oscillators in ensembles 1 and 2 for the three phase-synchronous frustration intervals. (a – c) correspond to the frustration range (0.1, 1.225), (d – e) (2.834, 3.549), and (g – i) (5.084, 5.780) with $m = 20$. (a, d, g) compare the phases of the two ensembles when they are uncoupled and coupled, (b, e, h) coupled and decoupled, and (c, f, i) uncoupled and decoupled.

allow synchronisation and strong restoration of the original phase configuration after decoupling.

These findings indicate that specific frustration values enable two ensembles to synchronise while preserving significant aspects of their memory-encoding phase structures. Such regimes therefore provide the strongest support for the hypothesis that mental synthesis may occur through synchronisation of memory-encoding neuronal ensembles.

Although this section focuses on phase-pattern retention, these steady states are

equivalently characterised by their synaptic weight configurations. As established in Section 4.1, the synaptic weight structure is uniquely determined by the phase relationships. Hence, the retention of phase patterns implies the retention of the corresponding synaptic memory maps.

In summary, this chapter demonstrates that the Kuramoto–Sakaguchi model with STDP and random coupling can synchronise independently formed ensembles while preserving memory characteristics for specific parameter regimes. In particular, the frustration intervals $\alpha \in (0.25, 1.125)$ and $\alpha \in (5.084, 5.780)$ allow synchronisation with partial or substantial memory retention. These results support the plausibility of the hypothesis that mental synthesis arises from the synchronisation of neuronal ensembles that encode distinct memories.

Chapter 6

Discussion and Conclusion

The main objective of this study was to develop a minimal mathematical model of mental synthesis, which is defined as the formation of novel mental imagery through the integration of existing memories. At the neuronal level, this work adopts the hypothesis proposed by Vyshedskiy [1], which posits that mental synthesis arises when the LPFC synchronises neuronal ensembles that encode distinct memories.

To construct a mathematical model of mental synthesis, we first developed a more biophysically plausible model of memory-encoding neuronal ensembles. Building on the work of Maistrenko et al. [94] and subsequent extensions by Ansariara et al. [104] and Botha et al. [69], we introduced random, sparse coupling into a Kuramoto–Sakaguchi model with STDP. This modification allows neurons to form long-range connections, improving the biological plausibility of the network compared to all-to-all and nonlocal, cyclic topologies.

The random topology preserves the dynamical features observed in nonlocal, cyclic topologies, including phase-synchronised and chimera states, while extending the parameter regimes in which these states occur. Importantly, random coupling introduces qualitative changes in the system dynamics. Under nonlocal, cyclic coupling, chimera states exhibit the classical spatio-temporal structure with one coherent group of oscillators next to an incoherent group of oscillators, the coherent group all running at the same average frequency. In contrast, the random coupling produces an unstructured chimera state, where coherent oscillators are not spatially localised in one group but are instead situated in multiple groups that are separated in space by the incoherent oscillators. Moreover, oscillators dynamically enter and leave the coherent subgroups

in an unpredictable manner.

The ranges of frustration supporting phase-synchronised and chimera states are also significantly expanded under random coupling. In particular, parts of the regions corresponding to full synchronisation in cyclic networks are replaced by phase-synchronised states, which have a non-trivial phase structure. This shift is important because non-trivial phase patterns can be different for different initial conditions and encode different synaptic configurations. Therefore, an increase in the region of phase synchronised states corresponds to an increased capacity of the model to represent distinct memory states.

Building on this framework, we constructed a network with two randomly coupled ensembles. Each ensemble was a Kuramoto–Sakaguchi model with the coupling strengths determined by a phase reduction-compatible STDP rule. We then introduced bidirectional inter-ensemble coupling, also governed by the STDP rule, to model the idea of mental synthesis. Our simulations showed that the two independent ensembles can be brought into synchrony by turning on the random inter-ensemble coupling. This was demonstrated by our observations of frequency entrainment across the two ensembles and an increased global coherence of the network with the two ensembles. Our findings support the feasibility of modelling mental synthesis as a synchronisation process between memory-encoding ensembles.

Crucially, synchronisation was shown to occur alongside the retention of memory characteristics. Analysis of phase-difference correlations revealed that, in appropriate parameter regimes, identifiable traces of the original phase patterns persist when the two ensembles are coupled and are substantially recovered after the ensembles are decoupled. This result has an important conceptual implication: the synchronised state does not overwrite the original memory structures but instead embeds them within a combined dynamical configuration. In this sense, the model captures a key requirement of mental synthesis, namely the integration of memories into a novel construct while preserving their underlying structure.

In particular, the frustration intervals $\alpha \in (0.25, 1.125)$ and $\alpha \in (5.084, 5.780)$ were found to support both synchronisation and significant pattern retention. These regimes therefore represent parameter regions in which the model most closely aligns with the proposed mechanism of mental synthesis. By contrast, other regions, such as $\alpha \in (2.834, 3.549)$, support synchronisation without retention, highlighting that synchronisation alone is insufficient for modelling mental synthesis.

Overall, our findings suggest that the combination of distributed connectivity and phase-based plasticity may provide a viable minimal framework for modelling mental synthesis. More broadly, they support the plausibility of the hypothesis that mental synthesis arises from the synchronisation of neuronal ensembles that encode distinct memories.

Despite these contributions, however, several limitations remain. The present study focused on phase-synchronised states and did not assess the synchronisation of ensembles in the chimera states. Although the chimera states may be more biologically realistic, their coupled ensemble behaviour was not analysed because of their non-stationary phase structure. Future work should investigate whether meaningful notions of memory retention can be defined for states with dynamically evolving phase patterns.

One important advantage of the model that we have presented is that it does not require any specific neuron as a building block for the ensembles that we have studied. In this sense, it is quite general and perhaps suggests that the phenomena of mental synthesis may not depend on the details of the biophysical properties of the underlying neural population, but rather on how the coupling strengths change when there is any form of neuronal spiking. When the two ensembles are evolved separately, then coupled to form one phase synchronous group, and then decoupled again, each ensemble returns to its original phase synchronous state and, importantly, retains a significant amount of its original character, i.e. there is not too much degradation of the original images from which the mentally synthesized image was constructed. This is a remarkable hysteresis-type effect, which occurs here in a relatively simple model with no explicit inertial terms.

One limitation of our current model is that it does not explicitly include a control ensemble to represent the LPFC, which is hypothesised to orchestrate inter-ensemble synchronisation. Incorporating a control mechanism that modulates coupling between ensembles would provide a more complete model of mental synthesis. Further, the study’s analysis was restricted to varying the frustration parameter α , with the potentiation and depression time constants and learning rate held constant. These parameters can be varied to investigate whether there may be additional parameter regimes with improved memory retention.

In addition, the simulations in this study were performed with the use of adaptive time-stepping methods and JIT-compiled functions to improve the efficiency of the computations and thereby enable the study of larger systems. Even with this optimisation, the system sizes we have studied are still much smaller than would be necessary to

model even the simplest biological system that could exhibit mental synthesis. Thus, our model has not yet been validated against any experimental data. Nevertheless, extending its framework to population-level models and comparing its predictions with neuroimaging results, such as those reported in [23], is a viable direction for future research.

In conclusion, this study has demonstrated that a Kuramoto–Sakaguchi model with STDP and random coupling can capture the key feature of mental synthesis, i.e., the ability to synchronise two independently synchronous neuronal ensembles while preserving the memories (images) that each individual synchronous ensemble originally encoded. Our results provide a foundation for further theoretical and empirical investigations into the neuronal mechanisms underlying mental synthesis.

Bibliography

- [1] Vyshedskiy, A.: On the Origin of the Human Mind. CreateSpace Independent Publishing Platform, North Charleston, SC (2014)
- [2] Vyshedskiy, A., Dunn, R.: Mental synthesis involves the synchronization of independent neuronal ensembles. *Research Ideas and Outcomes (RIO)* **1**, e7642 (2015). DOI 10.3897/rio.1.e7642
- [3] Pearson, J., Naselaris, T., Holmes, E.A., Kosslyn, S.M.: Mental imagery: Functional mechanisms and clinical applications. *Trends in Cognitive Sciences* **19**(10), 590–602 (2015). DOI 10.1016/j.tics.2015.08.003. URL <https://doi.org/10.1016/j.tics.2015.08.003>
- [4] Finke, R.A., Slayton, K.: Explorations of creative visual synthesis in mental imagery. *Memory & Cognition* **16**(3), 252 – 257 (1988). DOI 10.3758/BF03197758
- [5] Pearson, D.G., Logie, R.H.: Effects of stimulus modality and working memory load on mental synthesis performance. *Imagination, Cognition and Personality* **23**(2), 183 – 191 (2003). DOI 10.2190/KRQB-0CED-NX6J-HQ72
- [6] Pearson, D.G., Logie, R.H., Gilhooly, K.J.: Verbal representations and spatial manipulation during mental synthesis. *European Journal of Cognitive Psychology* **11**(3), 295–314 (1999). DOI 10.1080/713752317. URL <https://doi.org/10.1080/713752317>
- [7] Eardley, A.F., Pring, L.: Spatial processing, mental imagery, and creativity in individuals with and without sight. *European Journal of Cognitive Psychology* **19**(1), 37–58 (2007). DOI 10.1080/09541440600591965
- [8] Brandimonte, M.A., Hitch, G.J., Bishop, D.V.M.: Manipulation of visual mental images in children and adults. *Journal of Experimental Child Psychology* **53**(3), 300–312 (1992). DOI 10.1016/0022-0965(92)90041-4

- [9] Anderson, R.E., Helstrup, T.: Visual discovery in mind and on paper. *Memory & Cognition* **21**(3), 283–293 (1993). DOI 10.3758/BF03208267
- [10] Sagone, E., De Caroli, M.E., Falanga, R., Indiana, M.L.: Mental synthesis and creative thinking in typically developed italian children. *New Trends and Issues Proceedings on Humanities and Social Sciences* **7**(1) (2020)
- [11] Vyshedskiy, A.: Imagination in autism: A chance to improve early language therapy. *Healthcare* **9**(63) (2021). DOI 10.3390/healthcare9010063. URL <https://doi.org/10.3390/healthcare9010063>
- [12] Braverman, J., Dunn, R., Vyshedskiy, A.: Development of the mental synthesis evaluation checklist (msec): A parent-report tool for mental synthesis ability assessment in children with language delay. *Children* **5**(5), 62 (2018). DOI 10.3390/children5050062
- [13] Vyshedskiy, A., Mahapatra, S., Dunn, R.: Linguistically deprived children: Meta-analysis of published research underlines the importance of early syntactic language use for normal brain development. *Research Ideas and Outcomes* **3**, e20696 (2017). DOI 10.3897/rio.3.e20696. URL <https://doi.org/10.3897/rio.3.e20696>
- [14] Vyshedskiy, A., Khokhlovich, E., Dunn, R., Faisman, A., Elgart, J., Lokshina, L., Gankin, Y., Ostrovsky, S., deTorres, L., Edelson, S.M., Ilyinskii, P.O.: Novel prefrontal synthesis intervention improves language in children with autism. *Healthcare* **8**(4) (2020). URL <https://www.mdpi.com/2227-9032/8/4/566>
- [15] Vyshedskiy, A., Radi, K., DuBois, M.C., Mugford, E., Maslova, V., Braverman, J., Piryatinsky, I.: Novel linguistic evaluation of prefrontal synthesis (leps) test measures prefrontal synthesis acquisition in neurotypical children and predicts high-functioning versus low-functioning class assignment in individuals with autism. *Applied Neuropsychology: Child* **11**(2), 99–114 (2022). DOI 10.1080/21622965.2020.1758700. URL <https://doi.org/10.1080/21622965.2020.1758700>. PMID: 32420749
- [16] Vyshedskiy, A., Venkatesh, R., Khokhlovich, E.: Representational drawing ability is associated with the syntactic language comprehension phenotype in autistic individuals. *medRxiv* (2024). DOI 10.1101/2024.07.26.24310995. URL <https://www.medrxiv.org/content/early/2024/07/30/2024.07.26.24310995>
- [17] Vyshedskiy, A.: Neuroscience of imagination and implications for human evolution. *Current Neurobiology* **10**(2), 89–109 (2019)

- [18] Lillo-Martin, D.C., Gajewski, J.: One grammar or two? sign languages and the nature of human language. *Wiley Interdisciplinary Reviews: Cognitive Science* **5**(4), 387–401 (2014). DOI 10.1002/wcs.1297. URL <https://doi.org/10.1002/wcs.1297>
- [19] Pfau, R., Salzmann, M., Steinbach, M.: The syntax of sign language agreement: Common ingredients, but unusual recipe. *Glossa: A Journal of General Linguistics* **3**(1), 1–46 (2018). DOI 10.5334/gjgl.511. URL <https://doi.org/10.5334/gjgl.511>
- [20] Rizzi, L., Cinque, G.: Functional categories and syntactic theory. *Annual Review of Linguistics* **2**, 139–163 (2016). DOI 10.1146/annurev-linguistics-011415-040827
- [21] Tylén, K., Fusaroli, R., Rojo, S., Heimann, K., Fay, N., Johannsen, N.N., Riede, F., Lombard, M.: The evolution of early symbolic behavior in *homo sapiens*. *Proceedings of the National Academy of Sciences* **117**(9), 4578–4584 (2020). DOI 10.1073/pnas.1910880117. URL <https://www.pnas.org/doi/abs/10.1073/pnas.1910880117>
- [22] Vyshedskiy, A., Khokhlovich, E., Dunn, R., Faisman, A., Elgart, J., Lokshina, L., Gankin, Y., Ostrovsky, S., deTorres, L., Edelson, S.M., Ilyinskii, P.O.: Novel prefrontal synthesis intervention improves language in children with autism. *Healthcare* **8**(4) (2020). DOI 10.3390/healthcare8040566. URL <https://www.mdpi.com/2227-9032/8/4/566>
- [23] Yomogida, Y., Sugiura, M., Watanabe, J., Akitsuki, Y., Sassa, Y., Sato, T., Matsue, Y., Kawashima, R.: Mental visual synthesis is originated in the fronto-temporal network of the left hemisphere. *Cerebral Cortex* **14**(12), 1376–1383 (2004). DOI 10.1093/cercor/bhh098. URL <https://doi.org/10.1093/cercor/bhh098>
- [24] Waltz, J.A., Knowlton, B.J., Holyoak, K.J., Boone, K.B., Mishkin, F.S., de Menezes Santos, M., Thomas, C.R., Miller, B.L.: A system for relational reasoning in human prefrontal cortex. *Psychological Science* **10**(2), 119–125 (1999). DOI 10.1111/1467-9280.00118
- [25] Siclari, F., Baird, B., Perogamvros, L., et al.: The neural correlates of dreaming. *Nature Neuroscience* **20**, 872–878 (2017). DOI 10.1038/nn.4545
- [26] Buzsáki, G.: *Rhythms of the Brain*. Oxford University Press (2006)

- [27] Gray, C.M., König, P., Engel, A.K., Singer, W.: Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature* **338**(6213), 334–337 (1989). DOI 10.1038/338334a0. URL <https://doi.org/10.1038/338334a0>
- [28] Singer, W., Gray, C.M.: Visual feature integration and the temporal correlation hypothesis. *Annual Review of Neuroscience* **18**, 555–586 (1995). DOI 10.1146/annurev.ne.18.030195.003011. URL <https://doi.org/10.1146/annurev.ne.18.030195.003011>
- [29] Engel, A.K., Fries, P., Singer, W.: Dynamic predictions: oscillations and synchrony in top-down processing. *Nature Reviews Neuroscience* **2**(10), 704–716 (2001). DOI 10.1038/35094565. URL <https://doi.org/10.1038/35094565>
- [30] Fries, P.: A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. *Trends in Cognitive Sciences* **9**(10), 474–480 (2005). DOI <https://doi.org/10.1016/j.tics.2005.08.011>. URL <https://www.sciencedirect.com/science/article/pii/S1364661305002421>
- [31] Sehatpour, P., Molholm, S., Schwartz, T.H., Mahoney, J.R., Mehta, A.D., Javitt, D.C., Stanton, P.K., Foxe, J.J.: A human intracranial study of long-range oscillatory coherence across a frontal–occipital–hippocampal brain network during visual object processing. *Proceedings of the National Academy of Sciences* **105**(11), 4399–4404 (2008). DOI 10.1073/pnas.0708418105. URL <https://www.pnas.org/doi/abs/10.1073/pnas.0708418105>
- [32] Kreiman, G., Koch, C., Fried, I.: Category-specific visual responses of single neurons in the human medial temporal lobe. *Nature Neuroscience* **3**(9), 946–953 (2000). DOI 10.1038/78868. URL <https://doi.org/10.1038/78868>
- [33] Gelbard-Sagiv, H., Mukamel, R., Harel, M., Malach, R., Fried, I.: Internally generated reactivation of single neurons in human hippocampus during free recall. *Science* **322**(5898), 96–101 (2008). DOI 10.1126/science.1164685. URL <https://doi.org/10.1126/science.1164685>
- [34] Reddy, L., Tsuchiya, N., Serre, T.: Reading the mind’s eye: Decoding category information during mental imagery. *NeuroImage* **50**(2), 818–825 (2010). DOI <https://doi.org/10.1016/j.neuroimage.2009.11.084>. URL <https://www.sciencedirect.com/science/article/pii/S1053811909012701>

- [35] Cichy, R.M., Heinzle, J., Haynes, J.D.: Imagery and perception share cortical representations of content and location. *Cerebral Cortex* **22**(2), 372-380 (2011). DOI 10.1093/cercor/bhr106
- [36] Klein, I., Dubois, J., Mangin, J.F., Kherif, F., Flandin, G., Poline, J.B., Denis, M., Kosslyn, S.M., Le Bihan, D.: Retinotopic organization of visual mental images as revealed by functional magnetic resonance imaging. *Brain Research. Cognitive Brain Research* **22**(1), 26–31 (2004). DOI 10.1016/j.cogbrainres.2004.07.006. URL <https://doi.org/10.1016/j.cogbrainres.2004.07.006>
- [37] Rao, A.M., DeHaan, R.D., Kahana, M.J.: Synchronous theta networks characterize successful memory retrieval. *Journal of Neuroscience* **45**(16), e1332242025 (2025). DOI 10.1523/JNEUROSCI.1332-24.2025. URL <https://www.jneurosci.org/content/45/16/e1332242025>
- [38] Saint Amour di Chanaz, L., Pérez-Bellido, A., Wu, X., Lozano-Soldevilla, D., Pacheco-Estefan, D., Lehongre, K., Conde-Blanco, E., Roldan, P., Adam, C., Lambrecq, V., Frazzini, V., Donaire, A., Carreño, M., Navarro, V., Valero-Cabré, A., Fuentemilla, L.: Gamma amplitude is coupled to opposed hippocampal theta-phase states during the encoding and retrieval of episodic memories in humans. *Current Biology* **33**(9), 1836–1843 (2023). DOI 10.1016/j.cub.2023.03.073. URL <https://doi.org/10.1016/j.cub.2023.03.073>
- [39] Vivekananda, U., Bush, D., Bisby, J.A., Baxendale, S., Rodionov, R., Diehl, B., Chowdhury, F.A., McEvoy, A.W., Miserocchi, A., Walker, M.C., Burgess, N.: Theta power and theta-gamma coupling support long-term spatial memory retrieval. *Hippocampus* **31**(2), 213-220 (2021). DOI <https://doi.org/10.1002/hipo.23284>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/hipo.23284>
- [40] Kennedy, M.B.: Synaptic signaling in learning and memory. *Cold Spring Harbor Perspectives in Biology* **8**(2), a016824 (2013). DOI 10.1101/cshperspect.a016824. URL <https://doi.org/10.1101/cshperspect.a016824>
- [41] Sridhar, S., Khamaj, A., Asthana, M.K.: Cognitive neuroscience perspective on memory: overview and summary. *Frontiers in Human Neuroscience* **17**, 217093 (2023). DOI 10.3389/fnhum.2023.1217093. URL <https://www.frontiersin.org/journals/human-neuroscience/articles/10.3389/fnhum.2023.1217093>

- [42] Langille, J.J., E., B.R.: The synaptic theory of memory: A historical survey and reconciliation of recent opposition. *Front. Syst. Neurosci.* **12**, 52 (2018)
- [43] Hebb, D.O.: *The Organization of Behaviour*. John Wiley and Sons, New York (1949)
- [44] Bi, G.Q., Poo, M.M.: Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience* **18**(24), 10464–10472 (1998). DOI 10.1523/jneurosci.18-24-10464.1998
- [45] Zhang, L., Tao, H., Holt, C., et al.: A critical window for cooperation and competition among developing retinotectal synapses. *Nature* **395**, 37–44 (1998). DOI 10.1038/25665
- [46] Markram, H., Lübke, J., Frotscher, M., Sakmann, B.: Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps. *Science* **275**(5297), 213 – 215 (1997). DOI 10.1126/science.275.5297.213
- [47] Abraham, W.C.: How long will long-term potentiation last? *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences* **358**(1432), 735–744 (2003). DOI 10.1098/rstb.2002.1222
- [48] Maity, S., Narayanan, S.N., Millis, R.M., et al.: Long-Term Potentiation (LTP): A simple yet powerful cellular process in learning and memory. *Clinical Schizophrenia & Related Psychoses* **15**(4), 1–6 (2021). DOI 10.3371/CSRP.MSNS.090821
- [49] Citri, A., Malenka, R.C.: Synaptic plasticity: Multiple forms, functions, and mechanisms. *Neuropsychopharmacology* **33**(1), 18–41 (2008). DOI 10.1038/sj.npp.1301559
- [50] Caporale, N., Dan, Y.: Spike Timing-Dependent Plasticity: A Hebbian learning rule. *Annu. Rev. Neurosci.* **31**, 25–46 (2008). DOI 10.1146/annurev.neuro.31.060407.125639
- [51] Song, S., Miller, K.D., Abbott, L.F.: Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neuroscience* **3**, 919 (2000)
- [52] Sjöström, P.J., Rancz, E.A., Roth, A., Häusser, M.: Dendritic excitability and synaptic plasticity. *Physiological Reviews* **88**(2), 769–840 (2008). DOI 10.1152/physrev.00016.2007. URL <https://doi.org/10.1152/physrev.00016.2007>

- [53] van Rossum, M.C.W., Bi, G.Q., Turrigiano, G.G.: Stable hebbian learning from spike timing-dependent plasticity. *Journal of Neuroscience* **20**(23), 8812–8821 (2000). DOI 10.1523/JNEUROSCI.20-23-08812.2000. URL <https://www.jneurosci.org/content/20/23/8812>
- [54] Morrison, A., Diesmann, M., Gerstner, W.: Phenomenological models of synaptic plasticity based on spike timing. *Biological Cybernetics* **98**(6), 459–478 (2008). DOI 10.1007/s00422-008-0233-1. URL <https://doi.org/10.1007/s00422-008-0233-1>
- [55] Gütig, R., Aharonov, R., Rotter, S., Sompolinsky, H.: Learning input correlations through nonlinear temporally asymmetric hebbian plasticity. *Journal of Neuroscience* **23**(9), 3697–3714 (2003). DOI 10.1523/JNEUROSCI.23-09-03697.2003. URL <https://www.jneurosci.org/content/23/9/3697>
- [56] Fiete, I.R., Senn, W., Wang, C.Z., Hahnloser, R.H.: Spike-time-dependent plasticity and heterosynaptic competition organize networks to produce long scale-free sequences of neural activity. *Neuron* **65**(4), 563–576 (2010). DOI <https://doi.org/10.1016/j.neuron.2010.02.003>. URL <https://www.sciencedirect.com/science/article/pii/S0896627310000917>
- [57] Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology* **117**(4), 500–544 (1952). DOI 10.1113/jphysiol.1952.sp004764. URL <https://doi.org/10.1113/jphysiol.1952.sp004764>
- [58] FitzHugh, R.: Impulses and physiological states in theoretical models of nerve membrane. *Biophysical Journal* **1**(6), 445–466 (1961). DOI [https://doi.org/10.1016/S0006-3495\(61\)86902-6](https://doi.org/10.1016/S0006-3495(61)86902-6). URL <https://www.sciencedirect.com/science/article/pii/S0006349561869026>
- [59] Nagumo, J., Arimoto, S., Yoshizawa, S.: An active pulse transmission line simulating nerve axon. *Proceedings of the IRE* **50**(10), 2061–2070 (1962). DOI 10.1109/JRPROC.1962.288235
- [60] Wang, X.J.: Neurophysiological and computational principles of cortical rhythms in cognition. *Physiological Reviews* **90**(3), 1195–1268 (2010). DOI 10.1152/physrev.00035.2008. URL <https://doi.org/10.1152/physrev.00035.2008>
- [61] Ermentrout, G., Terman, D.: *Mathematical Foundations of Neuroscience. Interdisciplinary Applied Mathematics*. Springer New York (2010)

- [62] Izhikevich, E.M.: Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting. The MIT Press (2006). DOI 10.7551/mitpress/2526.001.0001. URL <https://doi.org/10.7551/mitpress/2526.001.0001>
- [63] Brown, E., Moehlis, J., Holmes, P.: On the phase reduction and response dynamics of neural oscillator populations. *Neural Computation* **16**(4), 673–715 (2004)
- [64] Acebrón, J.A., Bonilla, L., Vicente, C.J.P., Ritort, F., Spigler, R.: The Kuramoto model: A simple paradigm for synchronization phenomena. *Reviews of Modern Physics* **77**, 137 (2005). DOI 10.1103/RevModPhys.77.137
- [65] Da Fonseca, J., Abud, C.: The kuramoto model revisited. *Journal of Statistical Mechanics: Theory and Experiment* **2018**(10) (2018). DOI 10.1088/1742-5468/aadb05
- [66] Kuramoto, Y.: Self-entrainment of a population of coupled non-linear oscillators. *Lect. Notes Phys.* **39**, 420–422 (1975). DOI 10.1007/BFb0013365
- [67] Brede, M., Kalloniatis, A.C.: Frustration tuning and perfect phase synchronization in the kuramoto-sakaguchi model. *Physical Review E* **93**(6) (2016). DOI 10.1103/PhysRevE.93.062315
- [68] Kuramoto, Y., Battogtokh, D.: Coexistence of coherence and incoherence in non-locally coupled phase oscillators. *Nonlinear Phenom. Complex Syst.* **5**, 380-385 (2002)
- [69] Botha, A.E., Ansariara, M., Emadi, S., Kolahchi, M.R.: Chimera patterns of synchrony in a frustrated array of Hebb synapses. *Frontiers in Computational Neuroscience* **16**, 888019 (2022). DOI 10.3389/fncom.2022.888019
- [70] Pikovsky, A., Rosenblum, M., Kurths, J.: Synchronization: A Universal Concept in Nonlinear Sciences. Cambridge Nonlinear Science Series. Cambridge University Press (2001)
- [71] Buck, J.B., Buck, E.: Biology of synchronous flashing of fireflies. *Nature* **211**, 562-564 (1966)
- [72] Strogatz, S.: Sync: How Order Emerges from Chaos in the Universe, Nature, and Daily Life. Hachette Books (2003)

- [73] Kumari, N., Dwivedi, S.: Fundamental Concepts of Synchronization: An Introduction: From classical to modern. *Resonance* **25**(4), 539–565 (2020). DOI 10.1007/s12045-020-0969-z
- [74] Glass, L., Mackey, M.C.: *From Clocks to Chaos: The Rhythms of Life*. Princeton University Press (1988). URL <http://www.jstor.org/stable/j.ctv173dzwt>
- [75] Strogatz, S.H.: Norbert Wiener’s brain waves. In: S.A. Levin (ed.) *Frontiers in Mathematical Biology*, pp. 122–138. Springer Berlin Heidelberg, Berlin, Heidelberg (1994)
- [76] Malkin, I.G.: *Some problems in the theory of nonlinear oscillations*, vol. 1. US Atomic Energy Commission, Technical Information Service, Oak Ridge, Tenn. (1959)
- [77] Winfree, A.T.: Biological rhythms and the behavior of populations of coupled oscillators. *Journal of Theoretical Biology* **16**(1), 15–42 (1967)
- [78] Kuramoto, Y.: *Chemical Oscillations, Waves, and Turbulence*. Chemistry Series. Dover, New York (1984)
- [79] Sakaguchi, H., Shinomoto, S., Kuramoto, Y.: Mutual entrainment in oscillator lattices with nonvariational type interaction. *Prog. Theor. Phys.* **79**, 1069 (1988)
- [80] Li, X., Qiu, T., Boccaletti, S., Sendiña Nadal, I., Liu, Z., Guan, S.: Synchronization clusters emerge as the result of a global coupling among classical phase oscillators. *New Journal of Physics* **21**(5), 053002 (2019). DOI 10.1088/1367-2630/ab1ad5
- [81] Strogatz, S.H.: From Kuramoto to Crawford: Exploring the onset of synchronization in populations of coupled oscillators. *Physica D: Nonlinear Phenomena* **143**(1–4), 1–20 (2000). DOI 10.1016/S0167-2789(00)00094-4
- [82] Sakaguchi, H., Kuramoto, Y.: A soluble active rotator model showing phase transitions via mutual entertainment. *Prog. Theor. Phys.* **76**, 576 (1986)
- [83] Deco, G., Jirsa, V.K.: Ongoing cortical activity at rest: Criticality, multistability, and ghost attractors. *The Journal of Neuroscience* **32**(10), 3366–3375 (2012)
- [84] Olmi, S., Navas, A., Boccaletti, S., Torcini, A.: Hysteretic transitions in the kuramoto model with inertia. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* **90**(4) (2014). DOI 10.1103/PhysRevE.90.042905

- [85] Botha, A., Eclerová, V., Shukrinov, Yu.M., Kolahchi, M.R.: Effects of frustrated interactions on synchronicity in electrical power grids, pp. 53–61. Springer International Publishing (2023). DOI 10.1007/978-3-031-27082-6_5. URL <https://doi.org/10.3389/fncom.2022.888019>
- [86] Bag, B.C., Petrosyan, K.G., Hu, C.K.: Influence of noise on the synchronization of the stochastic kuramoto model. *Phys. Rev. E* **76**, 056210 (2007). DOI 10.1103/PhysRevE.76.056210. URL <https://link.aps.org/doi/10.1103/PhysRevE.76.056210>
- [87] León, I., Muolo, R., Hata, S., Nakao, H.: Higher-order interactions induce anomalous transitions to synchrony. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **34**(1), 013105 (2024)
- [88] Skardal, P.S., Arenas, A.: Explosive synchronization and multistability in large systems of Kuramoto oscillators with higher-order interactions. In: F. Battiston, G. Petri (eds.) *Higher-Order Systems*, pp. 217–232. Springer International Publishing, Cham (2022). DOI 10.1007/978-3-030-91374-8_8
- [89] Breakspear, M., Heinemann, S., Daffertshofer, A.: Generative models of cortical oscillations: neurobiological implications of the Kuramoto model. *Front. Hum. Neurosci.* **4**, 1 (2010)
- [90] Ashwin, P., Timme, M.: When instability makes sense. *Nature* **436**(7047), 36–37 (2005). DOI 10.1038/436036b
- [91] Motter, A.E., Myers, S.A., Anghel, M., Nishikawa, T.: Spontaneous synchrony in power-grid networks. *Nature Physics* **9**(3), 191–197 (2013). DOI 10.1038/nphys2535
- [92] Leander, R., Lenhart, S., Protopopescu, V.: Controlling synchrony in a network of Kuramoto oscillators with time-varying coupling. *Physica D: Nonlinear Phenomena* **301–302**, 36–47 (2015). DOI 10.1016/j.physd.2015.03.003
- [93] Cumin, D., Unsworth, C.: Generalising the kuramoto model for the study of neuronal synchronisation in the brain. *Physica D: Nonlinear Phenomena* **226**(2), 181–196 (2007). DOI 10.1016/j.physd.2006.12.004
- [94] Maistrenko, Yu.L., Lysyansky, B., Hauptmann, C., Burylko, O., Tass, P.A.: Multistability in the Kuramoto model with synaptic plasticity. *Phys. Rev. E* **75**, 066207 (2007)

- [95] Seliger, P., Young, S.C., Tsimring, L.S.: Plasticity and learning in a network of coupled phase oscillators. *Phys. Rev. E* **65**, 041906 (2002)
- [96] Abrams, D.M., Strogatz, S.H.: Chimera states for coupled oscillators. *Phys. Rev. Lett.* **93**, 174102 (2004). DOI 10.1103/PhysRevLett.93.174102
- [97] Abrams, D.M., Strogatz, S.H.: Chimera states in a ring of nonlocally coupled oscillators. *Int. J. Bifurcation Chaos* **16**, 21 (2006). DOI 10.1142/S0218127406014551
- [98] Bi, G-Q., Poo, M-M.: Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neuroscience* **18**, 10464 (1998)
- [99] Wittenberg, G.M., Wang, S.S.H.: Malleability of spike-timing-dependent plasticity at the ca3-ca1 synapse. *Journal of Neuroscience* **26**(24), 6610–6617 (2006). DOI 10.1523/JNEUROSCI.5388-05.2006. URL <https://doi.org/10.1523/JNEUROSCI.5388-05.2006>
- [100] Pfister, J-P., Gerstner, W.: Triplets of spikes in a model of spike timing-dependent plasticity. *J. Neuroscience* **26**, 9673 (2006)
- [101] Ren, Q., Zhao, J.: Adaptive coupling and enhanced synchronization in coupled phase oscillators. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* **76**(1), 016207 (2007). DOI 10.1103/PhysRevE.76.016207
- [102] Niyogi, R.K., English, L.: Learning-rate-dependent clustering and self-development in a network of coupled phase oscillators. *Physical Review E* **80**(6), 066213 (2009)
- [103] Timms, L., English, L.: Synchronization in phase-coupled Kuramoto oscillator networks with axonal delay and synaptic plasticity. *Phys. Rev. E* **89**, 032906 (2014)
- [104] Ansariara, M., Emadi, S., Adami, V., Botha, A.E., Kolahchi, M.R.: Signs of memory in a plastic frustrated Kuramoto model of neurons. *Nonlinear Dynamics* **100**, 3685 (2020). DOI 10.1007/s11071-020-05705-4
- [105] Andreev, A.V., Maksimenko, V.A.: Synchronization in interacting networks of Hodgkin-Huxley neurons. In: D.E. Postnov, B.N. Khlebtsov (eds.) *Computational Biophysics and Nanobiophotonics*, vol. 12194, p. 121940K. International Society for Optics and Photonics, SPIE (2022). DOI 10.1117/12.2626379. URL <https://doi.org/10.1117/12.2626379>

- [106] Huang, W., Yang, L., Zhan, X., Fu, Z., Jia, Y.: Synchronization transition of a modular neural network containing subnetworks of different scales. *Frontiers of Information Technology & Electronic Engineering* **24**(10), 1458–1470 (2023). DOI 10.1631/FITEE.2300008. URL <https://doi.org/10.1631/FITEE.2300008>
- [107] Huang, W., Wu, Y., Ding, Q., Jia, Y., Fu, Z., Yang, L.: Achieving synchronization and chimera state of modular neural networks by using dynamic learning to adjust electromagnetic induction. *Nonlinear Dynamics* **113**(7), 6671–6693 (2025). DOI 10.1007/s11071-024-10435-y. URL <https://doi.org/10.1007/s11071-024-10435-y>
- [108] Watts, D.J., Strogatz, S.H.: Collective dynamics of ‘small-world’ networks. *Nature* **393**(6684), 440–442 (1998)
- [109] Huang, M., Xue, Y., Luo, H., Wang, Y., Tang, Y., Wen, Q.: Dynamics of a Kuramoto model with two-body and three-body interactions. *International Journal of Bifurcation and Chaos* **33**(2), 2350025 (2023). DOI 10.1142/S0218127423500256
- [110] Bronski, J.C., Carty, T., DeVille, L.: Configurational stability for the Kuramoto-Sakaguchi model. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **28**(10), 103109 (2018). DOI 10.1063/1.5029397. URL <https://doi.org/10.1063/1.5029397>