UNIVERSITY OF LIVERPOOL

Pattern Formations with Discrete Waves and Broadcasting Sequences.

by

Thomas Nickson

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Faculty of Science and Engineering
Department of Computer Science

February 2013

“Midway in the journey of our life I came to myself in a dark wood, for the straight way was lost. Ah, how hard it is to tell the nature of that wood, savage, dense and harsh – the very thought of it renews my fear!”

Dante Alighieri

UNIVERSITY OF LIVERPOOL

Abstract

Faculty of Science and Engineering
Department of Computer Science

Doctor of Philosophy

by Thomas Nickson

This thesis defines the Broadcasting Automata model as an intuitive and complete method of distributed pattern formation, partitioning and distributed geometric computation. The system is examined within the context of Swarm Robotics whereby large numbers of minimally complex robots may be deployed in a variety of circumstances and settings with goals as diverse as from toxic spill containment to geological survey. Accomplishing these tasks with such simplistic machines is complex and has been deconstructed in to sub-problems considered to be significant because, when composed, they are able to solve much more complex tasks. Sub-problems have been identified, and studied as pattern formation, leader election, aggregation, chain formation, hole avoidance, foraging, path formation, etc. The Broadcasting Automata draws inspiration from a variety of sources such as Ad-Hoc radio networks, cellular automata, neighbourhood sequences and nature, employing many of the same pattern forming methods that can be seen in the superposition of waves and resonance.

To this end the thesis gives an in depth analysis of the primitive tools of the Broadcasting Automata model, nodal patterns, where waves from a variety of transmitters can in linear time construct partitions and patterns with results pertaining to the numbers of different patterns and partitions, along with the number of those that differ, are given. Using these primitives of the model a variety of algorithms are given including leader election, through the location of the centre of a discrete disc, and a solution to the Firing Squad Synchronisation problem. These problems are solved linearly.

An exploration of the ability to vary the broadcasting radius of each node leads to results of categorisations of digital discs, their form, composition, encodings and generation. Results pertaining to the nodal patterns generated by arbitrary transmission radii on the plane are explored with a connection to broadcasting sequences and approximation of discrete metrics of which results are given for the approximation of astroids, a previously unachievable concave metric, through a novel application of the aggregation of waves via a number of explored functions.

Broadcasting Automata aims to place itself as a robust and complete linear time and large scale system for the construction of patterns, partitions and geometric computation. Algorithms and methodologies are given for the solution of problems within Swarm Robotics and an extension to neighbourhood sequences. It is also hoped that it opens up a new area of research that can expand many older and more mature works.

Acknowledgements

I would like to thank, first and foremost, my supervisor, for always managing to make time and energy for me. My family for their vague interest, I couldn't have asked for more. My friends for providing constant distractions and inspirations without which I doubt I would have made it.

Contents

Chapter 1

Introduction

1.1 Background and Motivation

In many cases it is deemed that large numbers of simple robots can achieve certain tasks with greater efficiency than a single complex robot. Without the reliance on a single automaton there is an increase in robustness, the ability of the system to function with partial failures or abnormal conditions; flexibility, the capability of the system to adapt to changes in the environment; scalability, being able to expand the size of the system without a significant impact on performance [1, 2]. This leads to the question of how to co-ordinate these robots in their task whilst retaining their simplicity of design, function and the robustness that is inherent to distributed systems. Such robots may only have local information or minimal sensor range, small finite memory, minimal communication and restricted common knowledge such as coordinates, orientation and synchronicity [3, 4]. The body of work concerned with the design and analysis of systems with these properties and goals is commonly referred to as Swarm Robotics or more broadly distributive computations.

A variety of models have been suggested for this task including those from areas such as Multi Agent Systems, Engineering and Robotics, Cellular Automata (CA) and Distributed Algorithmics [5]. Problems currently being worked on are common subdivisions of more complicated and pragmatic tasks which are found to tackle a large set of problems simply through the coordinated use of such sub problems. These are often, but not restricted to, pattern formation [6], leader election [7], aggregation [8], chain formation [9], self assembly [10], coordinated movement [11], hole avoidance [12], foraging [13], path formation [14], etc.

Not only are there many different problems that need to be tackled in order to develop swarms of robots that are capable, in real world terms, but there are many ways of solving such problems, each of which may need different computational powers, and concepts, to solve them. In a survey of mathematical models of swarm robotics [15] four main areas were identified,

- **probabilistic models**, mostly used for foraging, chain formation, stick pulling, aggregation, cooperation-transportation, clustering, sorting, nest building and flocking;
- **automata theory** (which covers the use of cellular automata), mostly used for collaborative mapping, unknown exploration, lattice formation, pattern formation and stick pulling;
- **differential and partial differential equations**, mostly used for modelling pheromone, trophallaxis, stigmergic studies, collective foraging, clustering, flocking, schooling and attraction, alignment and repulsion systems;
- **dynamical systems** are largely used for cohesion and stability analysis, beacon and odour localisation, bifurcation models, attraction, alignment and repulsion systems, aggregation and population dynamics in swarms.

The survey paper [15] does not consider the use of models used primarily for distributed systems transferred over to use in swarm robotics. Such a method is employed in [16] where the authors propose a model, known as Parallel Mobile Robot (PMR), and are interested in models and algorithms for computational swarms, which describe in a step-by-step fashion how a collection of mobots can work together to solve specific computational problems. This model is proposed as an alternative to models which attempt to express swarm behaviours as shown by ants, bees and other swarms of insects. Mobots are relatively powerful with enough memory to store individual identities of the other mobots and knows their own position in the environment which is modelled as a two dimensional grid. Foundational problems in this model are tackled such as assigning identities to robots based on location, coordinated motion and parallel sorting. It is also noted that such a model may simulate a mesh and, as such, mesh algorithms are also applicable. The model allows a good notion of complexity with all algorithms analysed accordingly and it is also noted that there are other similar models, all with complexity analysis, that have been used to solve problems such as dispersal, separation, freeze tag problem, involving waking up a collection of robots and aggregation.

The Broadcasting Automata model takes a similar direction to the models that take their cues from distributed systems. It is hoped that this model may be used to construct deterministic algorithms with measurable complexity that are able to construct a wide variety of geometric algorithms for swarm robotics. For example, the algorithm design for the pattern formation problem in swarm robotics can often be complex and, sometimes, even impossible when the problem must be solved from any arbitrary configuration in a swarm where no information about the initial configuration and topology is available. The emphasis of preceding works, when tackling problems in pattern formation, have mainly been with heuristic algorithms while rigorous proofs of the correctness of the algorithms have not always been given [17].

Pattern formation may be defined as the coordination of a group of robots to get into and maintain a formation with a certain shape, such as a wedge or a chain. Current application areas of pattern formation include search and rescue operations, landmine removal, remote terrain and space exploration, control of arrays of satellites and unmanned aerial vehicles (UAVs). Previous methods of tackling the problem of pattern formation in swarm robotics can initially be divided in to two areas, that of centralised control and that of decentralised control [18]. As the notion of swarm robotics adapted for this thesis lies on the side of decentralised this will form the focus of the overview of previous literature.

There are many differing models used within the pattern formation literature although not all follow the same lines as the philosophy adopted here. In [19] pattern formation is achieved via a series of algorithms for squares, circles, polygons, but all robots are required to know the global positions of all the other robots and the algorithm used only permits analysis via simulation. In [20, 21] a method employing a notion gleaned from nature in the form of crystallisation of molecules is used. The process is similar to molecular covalent bonding and robots are afforded several bond sites, each with a particular marking, whereby varying the markings and the locations of such markings causes the generation of a variety of shapes. A similar process is suggested in [22] but in this case springs are used with the parameters of the virtual physics, which are noted to be difficult to discover, generate the distinct patterns on the plane. Graph theoretic approaches have been suggested in [23, 24] where they were used to construct control graphs which describes the control strategy (or behaviour) used by each robot, and the dependence of its trajectory on that of one or more of its neighbours. Pattern formation is determined by the construction of such control graphs and possible constructions for the graphs are given.

It is within all of these papers that some patterns emerge as to the limitations of such techniques. It is very complex to construct new shapes or describe the total set of all shapes that are generatable in these models. The models have unpredictable or convergent results which are shown through simulation. They require either complex reasoning and as such strong computational complexity for the robots, for example in the case of the spring simulation, or a complex set of sensors or both.

It was the intention of this body of work to develop new techniques towards a more deterministic approach to the field of swarm robotics allowing stricter guarantees on solutions with accompanying proofs of such guarantees by applying an alternative approach for pattern formation. The problem of pattern formation can be solved in to two stages:

1. to apply heuristic algorithms for forming a regular grid
2. to design deterministic algorithms on a swarm with a regular topology and local, non-oriented interaction

The proposal put forward in this work is to introduce a specific Broadcasting Automata model which makes use of I/O Automata arranged in a square lattice topology where communication is defined by broadcast to all automata within a certain radius. The choice of model was initially motivated by previous algorithms for the generation of triangular, square and hexagonal lattices by a swarm of autonomous mobile robots with limited sensor ranges which were studied in [6, 17, 25]. By extending the work within the swarm robotics idiom the validity of the Broadcasting Automata model becomes immediately apparent as augmenting already possible arrangements with partitioning and pattern forming abilities.

Arranging robots into a regular grid structure has a number of benefits in terms of self organisation and efficient communication. The choice of a lattice network topology and broadcasting within a certain range allows for simplistic and efficient communication compared to random topologies [26] that can be seen employed in other models for swarm computations [2].

There are three standard regular grids on the plane: triangular, square and hexagonal. This thesis will only consider the case in which the elements form a square grid, however, similar questions may be interesting to study in the triangular and hexagonal grid as an alternative to the square grid. The specific choice of square lattice for the model used in this thesis along with the use of networks of Moore machines compares strongly to the well known physical modelling tool, Cellular Automata (CA), a two (or more) dimensional mesh of finite state machines which are locally interconnected [27]. In CA cells are able to synchronously change states depending on the states of their neighbours depending on some local update rule.

The Broadcasting Automata model is homogeneous in each cells use of the same update rules mapping the states of its neighbours to a new internal state which is again available to its neighbours. The homogeneity of the update rules, the massive parallelism and local interactions also give a clear analogy to physical laws that govern the natural world, again, parallelling the system of broadcasting waves that is proposed by this thesis. The translations of the CA algorithms, in this case the Firing Squad Synchronisation Problem, in to the Broadcasting Automata model were not only for showing the links between the subject, but for the design of a specific algorithm to solve a geometrical problem in the Broadcasting Automata model. However, the fact that such a translation is possible illustrates that there is a close connection between the CA and BA models.

Cellular automata are notably employed as, powerful distributed computational engines, discrete dynamical system simulators, conceptual vehicles for studying pattern formation and complexity as well as original models of physics [28]. Cellular automata all for efficient parallel implementations of lattice models in nature and so are ideal for the analysis of many concurrent dynamical processes in nature. They also are ideal for modelling the global dynamic behaviour of local rule interactions. They have been employed in the simulation of any number of natural phenomena such as dendritic crystal growth, spatial patterns formed by reaction diffusion equations, self organisation in neural networks, turbulence in hydrodynamical systems as well as many others. They are also treated as abstract dynamical systems that can be used to study the complexity of the whole class of continuous and discrete systems and the quantification of complexity as a system property. Cellular automata can also be used to generate original models of fundamental physics based on local and digital processes which are computationally universal.

Such a model fits well with the criteria that is wished to be expressed in the Broadcasting Automata model. Cellular automata are able to concisely formulate distributed, parallel problems and simulations of physical phenomena and it has a well defined notion of complexity which has been deeply studied as an abstract formulation of the complexity of dynamical systems [27] and is notable in the computation of patterns and their analysis. However, such systems can be difficult to construct from simplistic rules due to the nature of cellular automata to be highly complex, like most dynamical systems, largely exhibiting patterns or trajectories that appear to be effectively random, or show signs of complicated structure in the form of long-range correlations, and in the sense that the ensemble of allowed patterns may be very hard to describe. Such elements of cellular automata go against the notions that are developed in the Broadcasting Automata model whereby a simple set of rules distributed to all automata are able to generate stable patterns which have a simple and well defined nature that naturally affords the construction of algorithms through the repeated application of pattern forming procedures. The complexity is easily calculated and, for all algorithms presented here, linear in time. This makes Broadcasting Automata a much more attractive proposal for the control of swarms of robots and indeed the construction of discrete geometric algorithms than Cellular Automata.

In the Broadcasting Automata model local communication is based on non-oriented broadcasting of messages to all of the automata in range, of the broadcast, is a key component of the model. Such 'blind' broadcasting (non-oriented) reduces the complexity of the model. This reduction in complexity is due to the fact that the network topology does not need to be kept track of through some complicated distributed system in order to promote robustness. Topology of the network is relegated to the physical transmission systems and placement of the transmitters that would be used in the physical realisation of the system that is suggested in this thesis. The very placement of the transmitters in the physical world constructs a network topology that is efficient. This notion closely resembles the field of ad-hoc networks [29] where they allow decentralised, self organising networks without the need for any fixed infrastructure broadcasting using wireless transmissions and routing handled in a node to node basis.

The proposed model, Broadcasting Automata (BA), is aimed at designing new mechanisms for pattern formation on a variety of lattices using basic transmission and computational primitives. Further, the BA is able to analyse the result of dissemination of information using a variety of broadcasting radii over a variety of lattices. An algorithm to elect the centre of a discrete disc is given to illustrate the power of Broadcasting Automata, and how they may be used to construct useful and novel algorithms, and solve a key problem in distributed systems known as leader election. The solution is also given in a very minimal setting with regards to the non-oriented broadcasting that is used in Broadcasting Automata. Other problems and areas such as discrete geometry, image processing and pattern formation shall also be augmented along the way.

The construction of Broadcasting Automata allows many parallels to be drawn with other areas of both mathematics, which shall be presented in the following paragraphs, and computer science, as has already been seen with its connections to distributed algorithms and cellular automata. Such disciplines have been more rigorously studied in the construction of algorithms and methodology than the ad-hoc methodology used in engineering and robotics papers and as such leads some of the way to realising the initial concept of the system to enhance swarm robotics with this rigour. The following section will hope to elucidate some of these parallels that are formed with the use of more rigid structures of other areas of mathematics which have been drawn as inspiration and providing background and motivation for the work on Broadcasting Automata.

The Broadcasting Automata model draws inspiration from natural phenomena from the following sources; waves, superposition, nodal patterns, normal modes, resonance, cymatics and moiré patterns. Many of these physical effects and phenomena are visible in the patterns of Chladni plates, see Figure. 1.1, such as resonance, nodal patterns, waves, superposition and normal modes. As such Chladni plates are a good starting place at expositing through visibility the these phenomena through modes of vibration on a rigid surface [30] and, in the original experiments, sand. The Chladni experiments were initially conducted by simply bowing a glass surface with sand on it. The vibrations generated by the bow would cause the surface, at resonance, to divide in to regions vibrating in opposing directions, positive motion in one region would be contrasted with negative motion in another. Such motion is divided by regions where there is none, with these points known as nodal lines. With sand, or flour as was originally used, sprinkled on the plate it groups in those areas where there is an absence of motion between the areas of vibration making them visible. It was thought that such a technique could be employed, simulated with automata transmissions across a structured topology to simulate the resonance that occurs in the rigid body, to solve a number of the common problems that occur in the construction of algorithms for swarms of robots such as leader election or the partitioning problem.

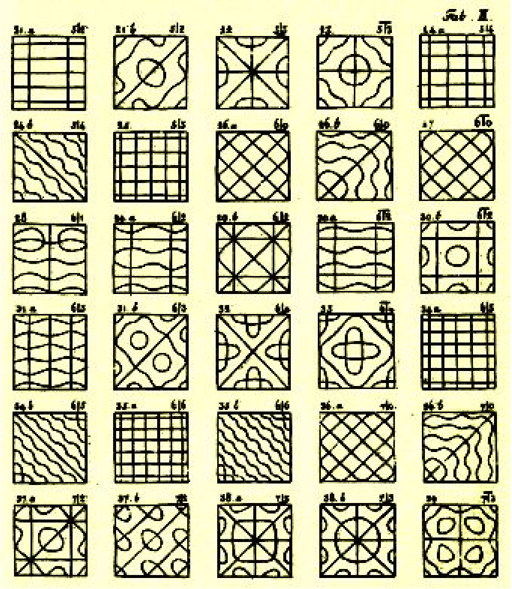


Figure 1.1: Showing a variety of modes present when varying the frequency of the source through a rigid body

The key features of the proposed algorithms, related to the simulation of information transmission through a network of automata, in the BA model are very similar to the techniques used in wave algorithms [31]. Such algorithms are so called because of the close resemblance to natural waves by analogy. One could imagine watching the formations generated by dropping a pebble in a square container of water and observing the waves generated spread through the medium and easily think of a simulation of this in wave in the virtual space of a square lattice network topology. Such algorithms are designed to solve very general problems found as subtasks in distributed algorithms; broadcasting of information such as start or terminate messages, synchronisation, initiating global events and computing function whereby each node holds some of the input. All such tasks are achieved by passing a finite number of message via some network topology dependent scheme such that all nodes participate in the message passing. Though such subtasks may seem simplistic they are able to solve complex problems such as leader election, termination detection and mutual exclusion and are used as subroutines to solve more involved algorithms.

Wave algorithms do not wholly encompass the Broadcasting Automata model that shall be presented here, in this thesis. Indeed, Broadcasting Automata are more general and the wave algorithms can be seen as a specific case of a Broadcasting Automata configuration, though naturally not all of BA can be explained by wave algorithms. The following notions are applicable to wave algorithms:

- Algorithms of this type may be centralised or decentralised depending on whether or not the algorithm is started by a single distinct node, known as a process, or can be started in all processes simultaneously.
- They may be designed for many prescribed topologies and may make use of initial knowledge in terms of a unique process identifier, access to its neighbours unique identifier and a sense of direction; none of which are present in broadcast automata.
- Each process makes at most a single decision after completion and the complexity may be measured in terms of the number of exchanged messages or bits for the computation.

Whilst the ideas represented here, in the above bullet points, as wave algorithms are well known and studied in the context of distributed computation the ideas presented in this thesis, under the name of Broadcasting Automata, which draw on the composition of successive waves, constructing nodal patterns, and indeed taking the analogy further than that presented in the classical notion, are novel and have not appeared previously in the literature. Whilst distributed computing has been used to model swarm computing it has not been so direct as to utilise wave algorithms and indeed wave algorithms have not been used to implement pattern formation or in the construction of wave inspired geometrical constructions.

The model for BA, which shall be presented in detail later in the chapter addressing the model, also draws from sources such as Ad-Hoc Networks [29] which is originally a model used for wireless cellular communication networks. Ad-Hoc networks are generally formed in situations in which a particular fixed infrastructure is not available. They are decentralised, self organising and capable of forming communication networks that do not rely on previously fixed structures. In this model nodes, elements within the network, are able to send and receive messages which are routed, in a decentralised fashion, through elements of the network from the node that generated the information to its recipient. Messages can be generated for any node in the network by any node in the network which allows a message to be generated in one node to be passed to any node in the network realising complete connectivity for every node. Such network design is a response to the increasing ubiquity of mobile computing and has many advantages over standard networking practice in that it increases mobility and flexibility, eliminates the need for a fixed infrastructure, increases robustness and reduces power consumption.

Ad-Hoc networks are a large and varied field of research with many differing models contained within its umbrella. Here, only those models which are deemed relevant to the thesis will be described. To this end pathloss geometric random graph models will be considered although it is noted that there are many other such models available such as random graphs, regular lattice graphs and scale free graphs. Geometric random graphs are considered to be more realistic when compared to the aforementioned models as they take in to account the geometric distance between nodes and many factors associated with the physical act of wireless transmission rather than associating connections between any two randomly chosen nodes. This model of transmission is based on a combination of empirical data collected on the characteristics of radio propagation and distilled in to an approximation of the workings of a radio channel commonly used for message passing in such networks through analytical methods. The assumption, for a message passed via radio signal to have been correctly received, is that it must be over a certain threshold value, $\mathcal{P}$. Those nodes that are able to receive messages from a transmitter are all nodes within the range $\mathcal{R} = r_0(\frac{c}{\mathcal{P}}^{\frac{1}{\eta}})$, where r_0 is a reference distance determined through empirical measurements (for example for low-gain antennas around 1-2 GHz is chosen to be 1m indoor and 100 - 1000 meters in an outdoor environment), η is the pathloss exponent which indicates the rate of the decay of the signal over distance and c represents the initial power of the transmission [32].

Using the above method, and varying $\mathcal{R}$, the graph that represents the network can be *dense* or *sparse*, *connected* or *disconnected*. The area of transmission of any node in the network is exactly the circle that is defined by such a radius, $\mathcal{R}$, and the probability of there being a connection between two nodes is exactly the probability of another node being within this circle of influence.

As introduced by Ad-Hoc networks in the geometric random graph model messages may be passed, under certain circumstances, to all nodes up to a certain range, $\mathcal{R}$. Such *repeated transmission* to nodes, within certain radii, applied to a certain network topology, or physical layout of nodes, in this case a number of geometric lattices, has been studied under the name, neighbourhood sequences (NS). Originally introduced in [33] under the notion of distance functions on digital pictures the aims were to produce algorithms that would define digital distance measures based on repeated applications of Moore and Von Neumann neighbourhoods, where each point within either neighbourhood is at distance one from the centre, which could be done in parallel using only local operations on every element (pixel) that composed a digitisation of a picture and its neighbour. Such algorithms could be used to solve problems such as, cluster detection, elongated part detection and regularity detection. The pictures were considered to be arrays (representing pixels for a display), which can be considered a square lattice structure underpinning the placement of wireless nodes, and the neighbourhoods were restricted to the Von Neumann and Moore neighbourhoods, which are equivalent to the digitisation of the circles of radii, 1 and $\sqrt{2}$, respectively, which naturally extends to the notion of $\mathcal{R}$.

Neighbourhood sequences were further explored in many more papers which again has found its main use in areas of indexing and segmenting images [34]. Extensions to the initial work include finding those distances which best approximate the euclidean distance [35–38] using both periodic and non-periodic sequences of the initial two neighbourhoods, Moore and Von Neumann, and their analogous extension to 3-dimensions [39–41]. From the exploration of such neighbourhoods the natural question of which of these form distance functions which adhere to the properties given to all metrics [42–44] has also been asked. Previous works have been extended to consider a variety of grids that extend the standard integer grid such as triangular [45, 46] and hexagonal grids [47–50]. In the analysis of the shapes that represent all of the possible digital discs (discretised discs that compose only integer points), an extension of the natural form of the neighbourhoods that made up the original NS's, they naturally form convex sets and as such may be combined using the Minkowski sum. In this way convex bodies and the Minkowski sum [51] are core parts of the analysis of Broadcasting Automata, as they give a method for composing convex sets in to new sets which are also convex, and as such derive the shapes of new Neighbourhood Sequences.

Much of the information derived from neighbourhood sequences is from observation of the geometric shapes, such as how close an approximation to the euclidean distance the sequence is, that can be generated by the iteration or repeated transmission of the sequences. This is analogous to the Minkowski sum of the sequences constituent discrete discs. Naturally this makes the Minkowski sum very useful in exploring the results of the combinations of digital discs with varying radii when generalising Neighbourhood Sequences. The analysis of such discrete discs also makes use of a method of digitising the hulls of such objects known as chain coding [52]. Chain codes for discrete discs can be constructed in a simple and efficient manor and, as will be proved, can make use of the notion of the Minkowski sum in their combination and manipulation to further the analysis of this extension to Neighbourhood Sequences.

Since the concept of composition has a number of limitations, most obviously noticeable is that of the resulting persistent convexity where all components are also convex, Broadcasting Automata afford new techniques derived from the discrete superposition of the waves, modelled as non-oriented transmissions of symbols. As such the concepts and formalisms addressed in the analysis of *moiré* phenomenon which concerns itself with the analysis of overlaid gratings and the resultant patterns that form from the alteration of the various parameters of both the gratings and the ways in which they are overlaid become increasingly useful in the analysis of the patterns resultant from Broadcasting Automata.

Moiré patterns [53] are the result of repetitive structures (gratings, screens, grids) with differing sizes or angles being superposed on top of each other. Originally the term *moiré* was derived from the common observation of the pattern in rippling water and its appearance in silk, which is itself essentially two orthogonal interlocking grids with varying and disparate distances between each strand. The patterns are as a result of the distribution of areas that are light or dark, analogous to the those areas that let light through or don't in binary patterns created by a grating. For example, in a monochromatic pattern, which is overlaid on top of itself, the interactions introduce a new pattern where one such structure may have removed a section that may have been translucent in the other and vice versa. *Moiré* phenomena has found use in a vast number of applications in a variety of different fields. In strain analysis *moirés* are used for the detection of slight deflections or object deformations where as metrology finds use for them in the measurement of small angles, displacements or movements. Other fields of use include optical alignment, crystallography and document counterfeiting as well as simply the appreciation of the variety of shapes that can emerge for fun or art.

Not only can moiré patterns be employed to help certain fields, for a variety of reasons but also they are, in many cases, a hindrance. In this case efforts are directed towards the elimination of the resultant interference which is beneficial to the printing industries where dot-screens can be severely corrupted by *moiré* patterns. Much work has been done in this area with regards to the patterns formed on the continuous plane with results formed through the use of the Fourier transform and the geometry of numbers as well as many other techniques, however, there is little in the way of results for the discrete plane. Many of the continuous results, if not the techniques, are still valid in the discrete plane and so many of the conclusions may be carried over. Analysis of *moiré* lines looks at differing models of superposition both of which employ some form of functional composition for example either multiplicative or additive, where intensity of the light source is modelled by sine functions, and it is within this method that the model presented here is taken further than in the realms of physics.

Broadcast Automata allows an increasingly flexible way of generating compositions, indeed the only limit is the number of compositions that are possible given a finite set and even that restriction is self imposed. The two basic functional compositions are given a treatment here but also other forms of composition are as well. As every discrete disc is a convex polygon which is composed of edges that are all straight lines with rational gradients only the basic *moiré* lines formed by such discrete objects need be considered but, as will be shown, this is a very powerful tool that can be used to relax many of the constraints on Neighbourhood Sequences.

1.2 An Overview of the Thesis

Here is presented a brief overview of the contents of the thesis.

In chapter 2 the model of Broadcasting Automata is introduced and the features and specifics are explored such as how and to whom messages shall be passed as well as an introduction to abstractions that concern such message passing. This chapter will prepare the ground work for the rest of the thesis presenting definitions of all the notions and notation present including a rigorous treatment of the connection to Neighbourhood Sequences as they are expanded to fit in with the work presented here and relabelled as Broadcasting Sequences. The main result in this section is the construction and definition of the model.

The third chapter covers an in depth look at the concept of nodal patterns, where waves, in a discrete sense, here modelled as words, may combine to form patterns which in turn generate partitions and, with the partitions, the generation of lines.

This chapter is the underpinning of all that follows it, giving an insight and elucidating what it is possible to accomplish with the model and how algorithms may be constructed. The results here are given as an exposition of what can be achieved with even the most basic components of the Broadcasting Automata model, the von Neumann and Moore neighbourhoods. There are a complete set of results on counting the number of distinct nodal patterns that result on the plane for 2-transmitter configurations and where waves are modelled as words.

Chapter 4 illustrates the power of the primitives further by showcasing a number of algorithms which can be performed in this model. Some of these algorithms are equivalents of algorithms found in a variety of other models such as Cellular Automata and distributed computing which here are the Firing Squad Synchronisation Problem, from the Cellular Automata problem set, and an algorithm for the election of centre of a circle which can be dually posed as a leader election algorithm, one of the most basic of problems from distributed algorithms as well as being posed as a distributed geometric problem. Also included in this chapter are a set of 'helper' algorithms which, algorithmically, extend the power of the model allowing the accomplishment of tasks that are thought to be common to a multitude of other algorithms that may be constructed. The main result of this chapter is the algorithm for the location of the centre of the digital disc where the initial transmitter is located on the edge of the disc. The algorithm uses all of the techniques discussed in the previous chapter, that utilise nodal patterns, to locate the centre of the disc through distributed, geometric computations.

The generalisation of Broadcasting Automata model, by varying the transmission radius, leads to the problem of the categorisation of digital discs. Results here include the form of the chain code of the discrete disc, how it may be generated and manipulated, and a restriction on their form. The structure of the discrete disc is of paramount importance, making up a large part of the potential power of the Broadcasting Automata model.

Chapter 6 gives results pertaining to nodal patterns that can be formed from arbitrary neighbourhood sequences and ways of overcoming the limitations in the inherent structures of the discs. This is given along with their connection with Neighbourhood Sequences which Broadcasting Sequences aim to generalise. This chapter primarily builds on the previous categorisation of the discs themselves focusing on how these structures may be composed and aggregated. Results include a notion of composability with a linear time algorithm such that any two discs, or indeed any two convex polygons that have been reduced to a chain code, may be composed to result in a single chain code that generates the new polygon. Further, the notion of composability that was first described as nodal patterns is explored with the resultant patterning of moiré and anti moiré functions categorised fully and many others shown experimentally. The aggregation of discs is then used to give experimental results on the approximation of the concave metric, the astroid, which was previously impossible to approximate with neighbourhood sequences.

Finally, chapter 7 presents the closing statements of the thesis.

The main body of this work has been presented in numerous places and formats. Chapters two through to four have been initially presented, in a talk, given at the *British Colloquium of Theoretical Computer Science*, first in 2009, under the title *A New Approach for Automata Coordination on* Z^2 and the following year a further expanded talk was given in 2010, under the title, *Geometric Computations on Regular Tilings by a Graph Dynamical System.* The results for these early chapters (2 to 4) were first published in *Geometric computations by broadcasting automata on the integer grid. Russell Martin, Thomas Nickson and Igor Potapov, volume 6714 of LNCS, pages 138-151* [54] with a talk given at the respective conference *Unconventional Computation 2011.* Further, expanded material was presented as a journal version in *Geometric computations by broadcasting automata. Russell Martin, Thomas Nickson and Igor Potapov, Natural Computing, pages 1-13* [55]. Finally, chapters 5 and, partially, 6, were accepted as a long abstract in *Discrete Discs and Broadcasting Sequences. Thomas Nickson and Igor Potapov, volume 7445 of LNCS, page 235* [56] with the presentation of a poster regarding the work given at the respective conference, *Unconventional Computation and Natural Computation 2012.*

Chapter 2

Broadcasting Automata and their Communication Primitives

Broadcasting automata connect many of the techniques contained in distributed algorithms, ad-hoc radio networks and Neighbourhood Sequences. Much like cellular automata with variably defined neighbourhoods Broadcasting Automata are defined usually on some form of grid or lattice structure and have a simple computational power comparative to a finite state automata with the ability to receive and send messages both from and to those automata which are within its transmission radius. Neighbourhoods are defined in the same way as with an ad-hoc network, all those points within a certain transmission radius, r, receive the message from the sender. Algorithms are in the same vain as those encountered in distributed algorithms using a simple notion of waves, messages passed from automata to automata throughout the topology, to construct computations. Wave algorithms are enhanced further with notions of composition of the information that is carried within each wave borrowed from the physical world and embellished with the new found computational power of the automata. The following sections shall solidify the notions that have been put forth here and give a concrete definition of the framework used to reason about computations constructed using waves, variable transmission radii and automata.

2.1 Broadcasting Automata Model

One of the fundamental models of computation is the automaton. The model can be used to explore the notion of the limitations and capabilities of computers and as such is useful in composing such models. They can be used to model digital devices that have very limited computational ability and limited memory. This makes automata a good starting point for the construction of a model for swarm robotics where limited computational power and memory are both aspirations of swarm robotics. Finite state automata take as input a word, which may in some instances be referred to as a tape, and output is limited to an accepting state which the automata is left in if it is said to accept the word. Traditionally automata used in cases where it is important to transform some input in to an output, beyond the use of a single state, is the use of Moore Machines. Such machines differ from finite state machines in that they are able to produce an output word from an input word.

Definition 2.1. A Moore machine [57] which is a 6-tuple,

$$A = (Q, \Sigma, \Lambda, \delta, \Delta, q_0) \ ,$$

where,

- Q is a finite set of states,
- Σ is the set of input symbols,
- Λ the set of output symbols,
- $\delta : Q \times \Sigma \to Q$ is the transition function mapping a state $q \in Q$ and a symbol, or set of symbols, $\sigma \in \Sigma$ to a state $q \in Q$,
- $\Delta : Q \to \Lambda$ is the output function which maps a state, $q \in Q$, to an output symbol, $\lambda \in \Lambda$, and
- q_0 is the initial or quiescent state in which the automata starts.

It is assumed that such a machine is connected to an input tape, or word, and an output tape, or word, where the result of the computation is read. In a situation, as is presented in distributed systems, whereby automata are connected to each other it is possible for the output of one automata to become the input of another automata. Such a model is known as a network of automata and connections from one automatons output to another's input may be represented as a directed graph, where direction represents the output going to input from automaton to automaton.

Definition 2.2. A network of finite automata is represented by a triple,

$$(G, A, C_0) \ ,$$

where,

- $G = (V, E)$ is a directed graph, with vertices, V, and edges, E, which are ordered pairs of vertices,
- A, is a Moore machine, and
- C_0 is an initial configuration which maps states, Q, of the automata, A, to vertices, V, such that, $C_0 : V \to Q$.

In this model the topology is fixed as specified by the construction of the graph, G, which dictates the flow of inputs and outputs from the Moore machines. Such symbols, where a symbol is part of the input/output word of the Moore machine, are generated as response to some input, by an automaton at vertex, $v \in V$, and then sent to all of the adjacent vertices in the graph or to a particular adjacent vertex, G, where the automata that receive the symbols process them as they would their input word.

This thesis introduces the model of Broadcasting Automata whereby nodes correspond to points in space and connectivity between nodes depend upon the distances between the points and the strengths of the transmissions generated by the automata as may be seen in ad-hoc radio networks. Transmission strength may vary from round to round for any particular automaton in the space and is dictated by the state of the automaton. In this model the topology, or connectivity graph, of the network of automata is able to change at each time step based on the states of the automata.

In order to give a formal definition of the Broadcasting Automata model on a metric space it is first necessary to introduce the notion of a metric space and to modify the classical notion of the Moore machine.

Definition 2.3. A metric space is an ordered pair, (M, d), where M is a set and d is a metric on M such that $d : M \times M \to \mathbb{R}$.

Here, (M, d), is any metric space, later the two dimensional euclidean space shall be used but this is not a necessity simply that a notion of the distance between two points is required.

Definition 2.4. The *Broadcasting Automaton* is represented by an 8-tuple

$$A = (Q, \Sigma, \Lambda, \delta, \Delta, \tau, q_0, F) \ .$$

This definition extends the Moore Machine by introducing a set of final states, F, along with a function, $\tau : Q \to \mathbb{R}$, which maps the state to a real number and represents the radius of transmission for the output symbol and the output alphabet, Λ, is extended by adding an empty symbol, ϵ.

A network of *Broadcasting Automata*, which will also be referred to as the *Broadcasting Automata model*, can now be defined.

Definition 2.5. The *Broadcasting Automata model* is represented by a triple such that,

$$BA = ((M, d), A, C_0) \ ,$$

where,

- (M, d) is a metric space,

- A is a Broadcasting Automata,
- C_0 is the initial configuration of the Broadcasting Automata model, it is a mapping from points, M, to states of the Broadcasting Automata, Q, such that, $C_0 : M \to Q$.

In some cases it may be that the input and output symbols are drawn from the same alphabet in which case, $\Sigma = \Lambda$.

The communication between automata is organised by message passing, where **messages** are symbols from the output alphabet, Λ, of the automata, A, to all of the automata within its transmission radius. Messages, symbols from the output alphabet, Λ, of the automata, A, are generated and passed instantaneously at discrete time steps, generation of message is given by the function, Δ, for the automata, A, resulting in synchronous steps. Those automata that have received a message, for the first time in the computation, are said to be **activated**.

If several messages are transmitted to an automaton, A, it will receive only a set of **unique messages**, i.e. for any multiset of transmitting messages, where the multiset represents a number of the same message being sent, received by A, over some number of rounds, the information about quantity of each type, within a single round, will be lost. This simply dictates that the automaton may not dictate a number of the same symbol in any transition, all transitions must operate upon a set of distinct symbols.

More formally the concept of computation and communication is captured in the concept of configurations of the Broadcasting Automata model and its semantics presented here.

Definition 2.6. The **configuration** of the Broadcasting Automata is given by the mapping, $c : M \to Q$, from points in the metric space to states. It can be noted here that C_0 is an initial configuration for the Broadcasting Automata model.

Automata are updated, from configuration to configuration, synchronously at discrete time steps. The next state of each automaton depends upon the states of all other automata which have in their neighbourhood the automaton that is to be updated. Where here the neighbourhoods are formed by a combination of their position in M and the range of their transmission, dictated by $\tau(q)$ for all automata.

Definition 2.7. The set of messages received by the automaton at a point, $u, v \in M$, is expressed by the set $\Gamma_u = \{\Delta(c(v)) | v \in M \wedge d(u,v) \leq \tau(c(v))\}$ for the metric space, (M, d).

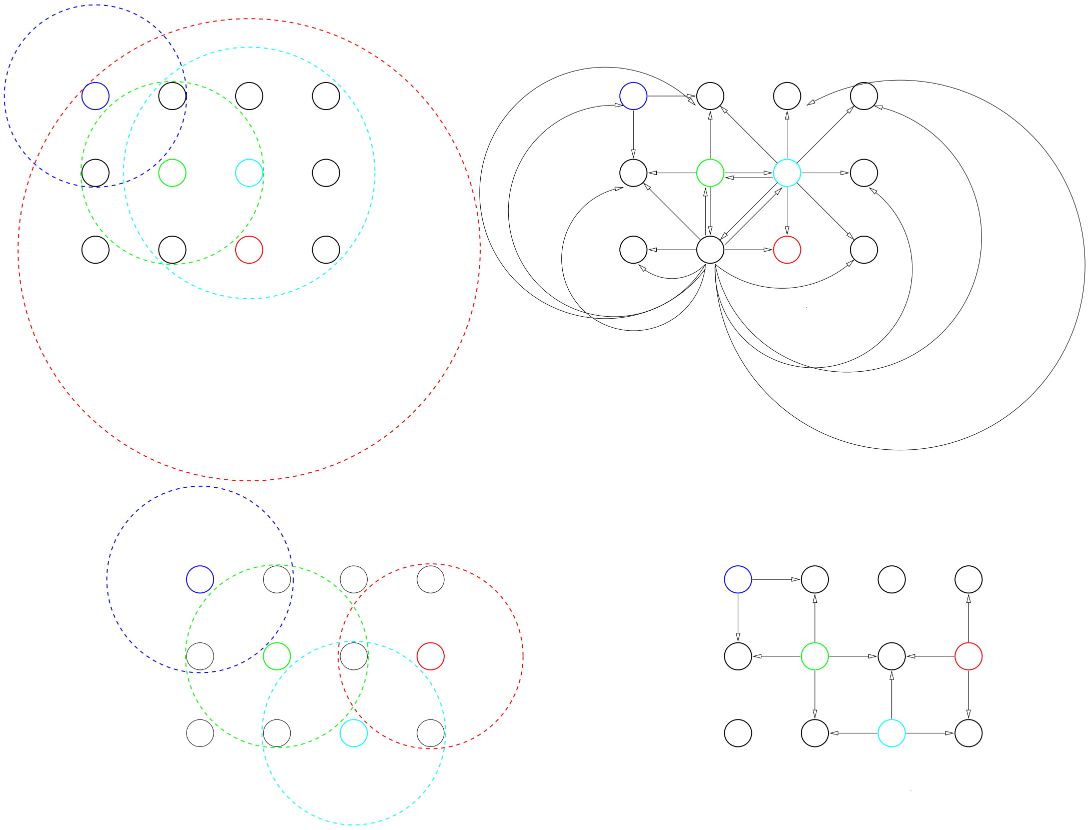


FIGURE 2.1: The above figures show the possible evolution over time of a network of Broadcasting Automata. Each figure shows, on the left, the broadcast range for the automata, depicted by a dotted circle with the node at the centre, on the right, the same connectivity is shown in graph form.

In Figure. 2.1, it is possible to see an elucidation of message passing in the Broadcasting Automata model through the process of constructing a digraph representation of the broadcasting radii. The figure depicts the automatons broadcasting radius by way of a dashed circle with the automaton broadcasting at that range shown in the centre. The corresponding graph may be constructed from this by making a directed edge in the graph from the node that is broadcasting to all nodes that are within the broadcast range, where the range is dictated by the state, $q \in Q$ and the function, $\tau(q) = r$ where r is the broadcasting radius. The construction of such a graph shows both the connection to the network automata model, which operates on a similar premise but with a fixed graph, but also highlights how Definition. 2.7 is constructed. Incoming edges are generated by connecting nodes, with the arrow from transmitter to node, reachable from the transmitter. Naturally such a graph, in the Broadcasting Automata model, can change over time which is shown by images further down the page as increase time steps in Figure. 2.1.

It should be noted that this set of messages can be empty. The new state of the automaton at u is now given by applying the automaton's transition function $\delta_u(\Gamma_u)$. This may be applied to all of the automata in the BA and as such determines the global dynamics of the system. It can be seen that in one time step it is possible for configuration c to become configuration e where, for all $v \in M$, $e(v) = \delta_v(\Gamma_u)$. Such a function is called the global transition function. The **set of all configurations for BA** is denoted as $\mathcal{C}$.

Definition 2.8. The global transition function, $\mathcal{G} : \mathcal{C} \to \mathcal{C}$, represents the transition of the system at discrete time steps such that for two configurations, $c \in \mathcal{C}$ and $e \in \mathcal{C}$, where e is the configuration the time step directly after c such that, $e = \mathcal{G}(c)$.

Definition 2.9. A computation is defined as any series of applications of the function, $\mathcal{G}$, from an initial condition, C_0, that leads to the set of all automata in the system being in one of their accepting states from the set, F, or the computation halts such that there exists no defined relation from the current configuration, c, under the global transition function $G(c)$.

Here a notion of reachability is given.

Definition 2.10. One configuration, c, is said to be **reachable** from another, e, if from the initial configuration, C_0, there exists a series of valid intermediary configurations such that $c \to c' \to c''... \to e$. Where $c \to c'$ is equivalent to

$\mathcal{G}(c) = c'$. It can also be stated, in a shorthand way, that where the sequence $c \to c' \to c''... \to e$ exists reachability from c to e may be shown as $c \rightsquigarrow e$.

With a concrete definition of a global transition function and a computation it is also possible to define what the computational complexity of an algorithm in the Broadcasting Automata model.

Definition 2.11. The **time complexity** of an algorithm is determined by the number of applications of the global transition function during the computation.

Whilst in general Broadcasting Automata may be used represent any of the common network topologies (such as, linear array, ring, star, tree, near-neighbour mesh, systolic array, completely connected, chordal ring, 3-cubes and hyper-cubes [58, 59]), through varying the location and transmission radii of the automata in the space , a different model is employed. The model used throughout the thesis is a hybrid between the notion of transmission radius in the pathloss geometric random graph model and common network topologies. The method involves the physical placement of nodes on the Euclidean plane in a lattice structure such that the distances between points conform to the euclidean distance. Such regular lattices coincide with those structures that can be formed by previous works [6, 17, 25] giving the basis of the forms of lattice that can be studied. This thesis will strictly concern itself with the **square grid lattice** though it can be extended in to any of the other grid configurations such as the hexagonal or triangular lattices along with many others.

The locations of the points on the lattice allow the construction of the metric space, (M, d) as such the locations of the automata. Assuming a lattice structure for positioning of the automata and transmission radii, $\mathcal{R}$, is in accordance with the principles of transmission used in the pathloss geometric random graph model put forth in Ad-Hoc networks.

A variety of topologies, the lattices that can be constructed as in [6] (square, hexagon and triangle), and the shapes of neighbourhoods generated by varying the transmission neighbourhood can be seen in Figure 2.2 where black dots represent the placement of automata on the plane according to the lattice here shown by the black lines. Neighbours are those within the dashed circle and the automata at the centre of the circle is the initial transmitter of any messages.

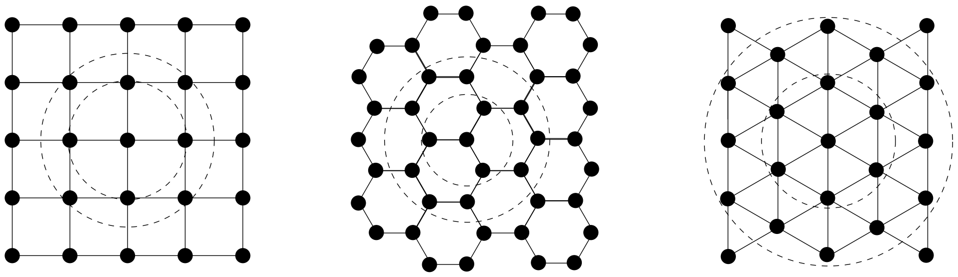


FIGURE 2.2: Showing three differing lattices and how their neighbourhoods can be altered simply by varying the radii of the transmission neighbourhood which, here, is depicted as a dotted circle. All automata, shown as black dots, are able to receive messages from the automata at the centre of the circle.The three lattices depicted are (left to right) square, hexagon and triangle.

The connectivity graph defined in ad-hoc radio networks denotes those that are able to send and receive messages with certainty however in distributed algorithms the possibility of communication errors is not ignored. In such models messages may be lost, duplicated, reordered or garbled where they must be detected and corrected by supplementary mechanisms which are mostly referred to as protocols for example the common network protocol of TCP. The Broadcasting Automata model sides with the former method of message passing. It is assumed that there will be no error in transmission, the receipt or sending of messages across the network is guaranteed however consideration is made to the synchronicity of the network due to the high sensitivity to the timing of message passing as will become clear later when discussing pattern formation with such protocols. Usually, though not a requirement of the model, structured sequences of messages are used such as a string where each of the symbols are indexed, here denoted as σ_i, showing that it is the ith message in such a structured string. This idea is explored fully later in the thesis, particularly chapter 3, and makes up a large body of the work exploring the variety of these strings and the constructs that they form in the system.

To this extent we consider two variants of Broadcasting Automata: **synchronous** and **asynchronous** (or reactive) models. In each of the variations both the transmission of a method and amount of time take for the automata to process the message is considered to be constant. It is this constant time that will afford synchronisation within the model. The two differing models are presented to show that there are possible complexity trade offs depending on how it is to be measured. If the size of the alphabet is a priority, such that it must be minimised, then the synchronous model is best, however this comes at a penalty of time complexity, where the asynchronous model fairs better.

In the **asynchronous** model upon the receipt of a message, an input symbol (or set of symbols) $\sigma \in \Sigma$, the automaton, A, becomes active. The automaton may only react to a non-empty set of messages, it may not make and epsilon transition. Once activated the automaton, A, reacts to the symbol(s), σ, according to the configuration of the automaton and responds with the transmission of an output symbol, $\lambda \in \Lambda$, to those in its transmission radius for the current state, $q \in Q$, $\tau(q)$. The processing of the input symbols by the automaton is done in one discrete time step which is the same period for all automata in the graph. Once active the automaton must wait for another message in order to change its state it is not allowed epsilon transitions.

The **synchronous** model has a singular alphabet and as such, $\Sigma = \Lambda$ with the allowance of epsilon transitions. Upon the receipt of a message, an input symbol (or set of symbols) $\sigma \in \Sigma$, the automaton, A, becomes active. Once active the automaton is synchronised with the rest of the graph by repeated epsilon transitions that are representative of a constant transition via input symbol as in the asynchronous model. As each such transition is considered a change in configuration of the automaton it must take a single time step. As such this guarantees the synchronisation of system. However the multiple alphabet can be simulated by associating different symbols to different time steps as shall be seen later.

The following outlines, informally, the methodology of message passing used by the two differing constructions of automata, synchronous and asynchronous.

In the **synchronous** model messages are passed from automaton to automaton according to the following rules:

1. An automaton, A, receives a message from an activating source at time, t;
2. At time $t+1$, A sends a message to all automata within its transmission radius dependent on its state, $q \in Q$, $\tau(q)$;
3. At time $t+2$, A ignores all incoming messages for this round.

In the **asynchronous model** the following rules can be applied:

1. An automaton, A, receives a message $\sigma_i \in \Sigma$ from an activating source at time t;
2. The automaton, A, broadcasts a message $\sigma_{(i+1) \mod |\Sigma|}$ to all automata within its transmission radius, dependent upon its state, $q \in Q$, $\tau(q)$ at time $t+1$;
3. Ignore all incoming messages at time $t+2$.

In both models Step 3 prevents an automaton from receiving back the message that has just been passed to the automatons neighbours by ignoring all transmissions received the round after transmission and ensuring that the messages are always carried away from the initial source of transmission. In both cases this rejection of all messages may be modelled by the addition of a state that simply does not accept input in that state, where epsilon transitions apply, or that the transition is made to another equivalent state independent of input received, where from here it is possible to receive transmissions that once again affect the logic of the program.

Naturally, given these two models it is interesting to see if they are capable of the same computations and that indeed anything that can be done in one model can be done in the other. There are many ways with which to establish equivalence between automata and models in general. One important technique that has been used to show equivalence in connection with Turing machines is **simulation** [60].

Here a similar proposition is made with respects to establishing the equivalence of the two models of automata that are given here, synchronous and asynchronous. The global configuration function doesn't change between the two models as the function shows only how messages are routed through the graph and how those messages cause the automata to update states.

In this case the differences between the two models are restricted to the construction of the automata themselves and a mapping between the two alphabets that differ in size. Automata in the asynchronous model are restricted to state changes only upon the receipt of a set of messages, as epsilon transitions are not allowed, and the size of the output alphabet, $|\Lambda|$, is unbounded. Automata in the synchronous model can make epsilon transitions, as such they may make state transitions without any input, but they only have an input and output alphabet of size one, $|\Sigma| = |\Lambda| = 1$.

Usually when equivalence must be shown for two automata it is done by showing that they accept the same language [60], however, this is not possible as the two alphabets are, by definition, different, due to their cardinality. It is for this reason that the definition of equivalence, via simulation, must be altered such that the construction using multiple additional states is able to simulate the outcome of an automata with multiple additional members of its alphabet. It must be shown that it is possible to map one language in to another using nothing more than extra transitions of the automatons transition function in the synchronous model. Having shown that there is a possible mapping from the language of the asynchronous model, on a symbol by symbol basis, to a series of computations in the synchronous model, simulating an input of the symbol in the asynchronous model, it will be possible to construct automata that, using the mapping from language to automata construct are equivalent in their computations.

Slightly more formally it can be said that for every transition of the asynchronous model, $\delta_a(q_i, \sigma_i) = q_j$ there exists a simulation, utilising epsilon transitions, such that, $\delta_s(\delta_s(\delta_s(q_i, \epsilon), \epsilon)..., \sigma) = q_j$, where here σ is the only symbol in the synchronous alphabet.

Proposition 2.12. *Both the synchronous and asynchronous models are able to simulate the other.*

Proof. The following proof considers the input and output alphabets to be the same and as such $\Sigma = \Lambda$

Any algorithm for broadcasting automata in the asynchronous model with a message alphabet of size $|\Sigma|$ can be simulated by the synchronous model with a singular alphabet and $|\Sigma|$ slowdown.

Broadcasting a message $\sigma_i \in \Sigma$, such that σ_i is the ith message in Σ, using the reactive model with a message alphabet of size $|\Sigma|$ can be simulated in synchronous model by introducing a number of states equal to $|\Sigma|$. Once the automaton is activated with the initial activation message (the synchronous model only has a single alphabet) at time t the initiating automata waits until time $t + i$, after which i epsilon transitions have taken place and another message from the initiator indicates to stop epsilon transitions. The states are labelled such that the state after i epsilon transitions is equivalent to σ_i. Thus both end up in same state. Such a construction allows simulation in both directions to occur.

□

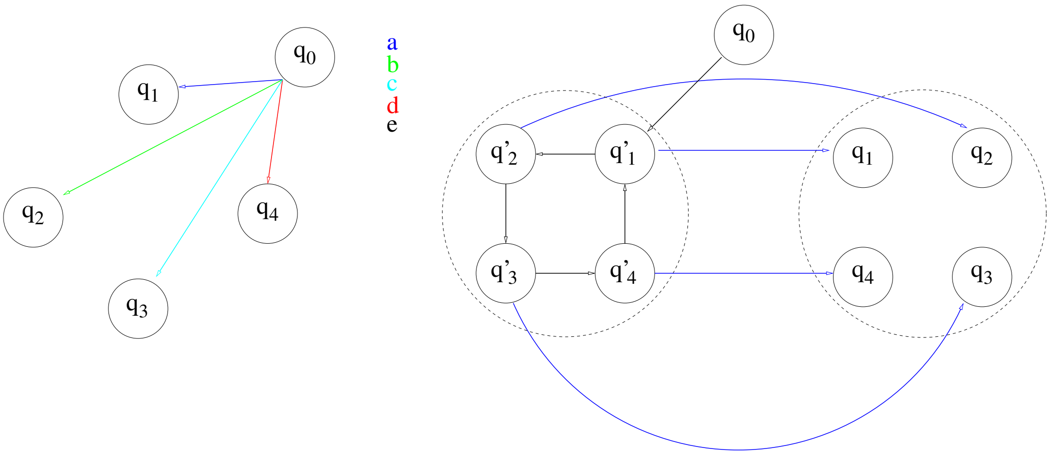


FIGURE 2.3: The above diagram depicts an equivalent, multi-alphabet, asynchronous automaton and a single alphabet, with epsilon transitions, synchronous automaton. Symbols, a, b, c, d and e, the epsilon transition, are colour coded such that receipt of the symbol a is signified by a blue edge.

An example of converting a basic construction in the asynchronous model in to a construction in the synchronous automata is depicted in Figure. 2.3. The automaton on the left of the figure depicts the asynchronous multi-alphabet model, where as the automaton on the right depicts the equivalent construction in the synchronous model where e represents the epsilon transition. Both automata attempt to recognise the input of a single letter of the alphabet such that the automata should end up in a state depending on the letter of the alphabet, $\Sigma = a, b, c, d$, or the time the symbol is received. It is noted that the method of transmitting the symbol in the synchronous model must also be encoded with a time delay such that when the automata sending the message wishes for the automata to end up in state q_i the sender must wait until the receiver is in state, q_i'.

For simplicity all examples that follow shall use the asynchronous model but it is simple to construct the corresponding synchronous automaton.

It should also be noted that with some generalisation, to allow a variable radius neighbourhood interaction which has been explored in a limited sense [61, 62], the Cellular Automata (CA) model may also be used to simulate the Broadcasting Automata model. In a grid of CA it is possible to assign states that correlate to transmitters where each transmitter in the grid is assigned a unique state and all other CA are in the quiescent state. The initial transmitters state causes each of the automata who have the initial transmitter in their neighbourhood to change from the quiescent state to the transmitters state plus one over some modulo. This clearly simulates the transmission of some word over a modulo for all automata in the grid. Upon finding a CA with one of this initial set of states within its neighbourhood, and such that it is all encompassing in its neighbourhood, the second transmitter changes its state which triggers a second cascade of state changes that are equivalent to those from the first transmitter only this time the automata must take in to account the initial state that it is already in. This should provide an exact copy of the Broadcasting Automata model, assuming that CA is extended to allow such variable neighbourhoods. This does not mean however that there is an exact translation of Cellular Automata in to Broadcasting Automata and as such it is only possible to say that $BA \subseteq CA$. Thus, the ability to translate the Firing Squad Synchronisation Problem still holds merit and should not be seen as a simple task such that any algorithm may find an equivalent in Broadcasting Automata.

Partial Vision

The broadcasting automata model is such that each element is only aware of its own state, and information about its neighbours is not directly observable which is, for example, contrary to the model of Cellular automata. However, it is possible to organise a **partial vision** which provides access to a set of distinct values of nodes from its neighbours via a non-oriented broadcast. In order to get this set of distinct values an automaton should send a special message (state request), $\lambda_{st} \in \Lambda$, to all neighbours which, in return, will send their state in the form of a message on the next step. Such a message is implemented in the automata by a reflexive relation on all states that upon receipt of the state request causes a reiteration of the symbol contained in the state, $\Delta(q)$, for $q \in Q$. It is possible to structure the automata such that an iterative process of such requests propagates through the whole network of broadcasting automata where each automata will react to this information by transmitting the value of its own state which naturally cascades through the system. This method of transmission omits the usual quiescent, state implemented within the automata, such that an automata is always open to the receipt of messages. The automata may also be structured in such a way that they ignore the state request message, in the same way that the usual construction of the message passing protocol, either synchronous or asynchronous, is formed.

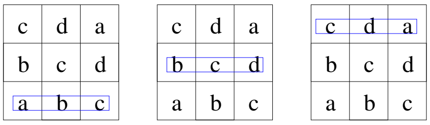
Partial vision is usually used to access a set of final states that represent a small portion of the completed ‘program’. Such states contain as their output message the information required when constructing computations with which to solve problems in this model which usually rely on the constructions of partitions using patterns which in turn are constructed by the information about the different states of the neighbourhood gleaned from broadcasting the state request message, σ_{st}. It may be that several subproblems need to be solved, and chained together one after another, in order to solve a problem. In algorithms that are constructed in chapter four of the thesis state requests of the neighbourhood are generally used to indicate to the automata which section of the solution to the previous subproblem they are in and as such what further action they need to take. It is in this way that a notion of 'programming' becomes evident within the Broadcasting Automata paradigm, though much more thorough examples of this will be given throughout chapter four.

Such notion of partial vision may be built up over time which is a process of the automata making the relevant transitions in accord with the symbols received, as with any other method of processing messages given here. In this way there is nothing special about the processing or the reaction of the receipt of the messages that make up the partial vision of the neighbourhood. Only the way in which state requests are instantiated, using the special symbol and the construction of the automata to accommodate such a message, differ from any previous definition.

As an illustration of the notion of partial vision in the model an example is given and shown in the Figure 2.4. Here the central vertex of the square will be considered, the box that progresses through the square shows the motion of the iterative messages from the surrounding automata (though this motion may be in many other directions) at each time step and the states, $\{a, b, c\} \in \Lambda$, represent categorisations of the neighbourhood. In Figure 2.4 the model that will be more thoroughly introduced later, in two dimensions and such that the automata are placed on the plane in the form a of a square lattice, is used along with the form of wave generate where all states broadcast at a radius of $\sqrt{2}$, $\tau(Q) = \sqrt{2}$.

In the first figure, i), the transmission to the central vertex will be the set $\{a, b, c\}$ and so this will be recorded as part of the neighbourhood. If the vertices were to transmit the characters a, b, a, replacing the c with a, then this would mean that the central vertex would record it's neighbourhood as $\{a, b\}$ for this time step. The transmission of the wave now progresses as shown in the second and third figures, the central vertex receiving first $\{b, c, d\}$ and then in the next time step

$\{c, d, a\}$. This constructs a set that is the neighbourhood of the vertex, in this case $\{a, b, c, d\}$.



i) ii) iii)

FIGURE 2.4: Showing three stages of neighbourhood recognition for the automata.

So far a complete account of the broadcasting and synchronisation properties has been addressed. The model described is weak (indeed it is shown to be the weakest model studied here [63]) in the sense that neither input nor output ports are labelled (it is non-oriented) and all automata send the same message to all neighbours and only a set of messages is receivable in return. The automata are also homogeneous in the sense that they all run the same algorithm, represented by their states. Such considerations are made to fit with the domain of the problem, swarm robotics, as previously discussed. Now, more specific considerations will be addressed with respects to the set of problems tackled and explored by this thesis which is by no means a complete exploration of the model presented here.

2.2 Variable Radius Broadcasting over $\mathbb{Z}^2$

Whilst the model provided here covers many configurations for the underlying communication graph this thesis shall only consider the following case where all automata are place in euclidean space according to a regular arrangement equivalent to that of a square lattice such that, using the Cartesian coordinate system, for any automata in the space its nearest neighbour is at distance 1 and its next nearest neighbour is at distance $\sqrt{2}$. This idea is illustrated for two dimensions in Figure. 2.5. This now leads to the model of Broadcasting Automata where (M, d) is $M = \mathbb{Z} \times \mathbb{Z}$ and where for $\rho = (\rho_0, rho_1)$ and $\rho' = (\rho'_0, \rho'_1)$ the distance function is $d(\rho, \rho') = \sqrt{(\rho_0 - \rho'_0)^2 + (\rho_0 - \rho'_0)^2}$, the Euclidean distance function for $\rho, \rho' \in M$.

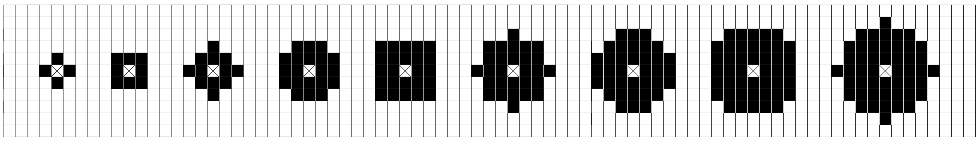


Figure 2.5: A variety of transmission radii are shown (l-r) squared radii $r^2 = \{1, 2, 4, 5, 8, 9, 10, 13, 16\}$. Crosses represent the centre of the respective discrete disc.

Figure 2.6 shows as example the automata that receive a message when the euclidean space and square lattice are restricted to two dimensions but the radius of broadcast, $\tau(q)$ for $q \in Q$, is varied. The automaton at point $\rho \in M$ which is the source of the transmission is shown as the circle at the centre of the surrounding automata on the plane, those that are black are within the transmission range $\tau(q)$ for $q \in Q$. Successive larger collections of automata coloured black show what happens when range can be changed to alter the automata that are included in the transmission radius, $\tau(q) \in \mathcal{R}$. If the transmission radius, $\mathcal{R}$, is equal to 1, as in Figure 2.6 diagram a) then only four of the eight automata can be reached. If the radius is made slightly larger and is equal to $\sqrt{2}$, it can encompass all eight automata in its neighbourhood as shown in diagram b). Such structures are identical to the well studied neighbourhoods von Neumann and Moore respectively and as such here it shall be considered that such constructions of neighbourhoods are a generalisation of these two neighbourhoods. As we will show later, iterative broadcasting within von Neumann and Moore neighbourhoods can distribute messages in the form of a diamond wave and a square wave as shown in Figure 2.6.

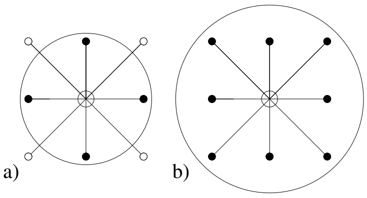


Figure 2.6: Diagram a) represents the propagation pattern for a diamond wave (Von Neumann neighbourhood) and diagram b) shows the propagation pattern for a square wave (Moore neighbourhood).

The construction of distinct radii for the circles is defined by the numbers n such that $n = x^2 + y^2$ has a solution in non-negative integers x, y [64]. Here it can be seen that r^2 is given for convenience as n always has a convenient representation in $\mathbb{Z}$ whereas it becomes cumbersome to write out either the root of the integer or its decimal representation. For an explanation as to how these numbers relate to the distinct discrete discs it must be noted that distinct discs are constructed such that r^2 contains a new solution in x and y. As it can be considered that the digital disc represents all of the maximal combinations of Pythagorean triples such that $x^2 + y^2 \leq r^2$ then for one disc to differ from another there must be an increase of r^2 such that there exists a new, distinct, maximal solution for $x, y \in \mathbb{Z}$. Generating these in the inverse direction by choosing $x, y \in \mathbb{Z}$ such that it is a new maximal combination not contained in any of the previous, smaller discrete discs allows the construction of the list of r^2 that defines the sequence of distinct discrete discs. This notion is illustrated in Figure. 2.7.

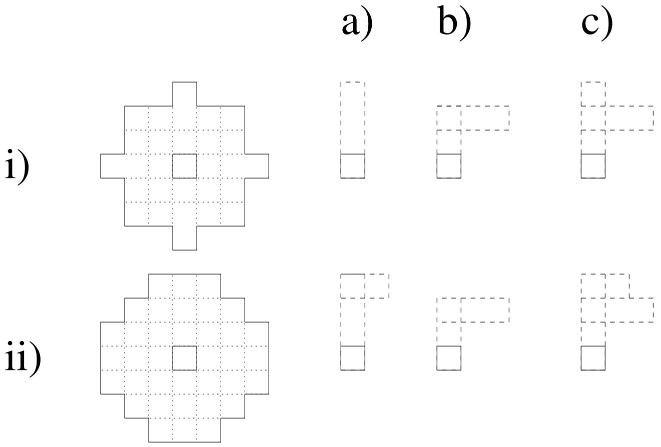


FIGURE 2.7: Showing two discrete discs i where $r^2 = 9$ and ii where $r^2 = 10$. The dashed figures beside each discrete disc, a and b, represent the two maximal combinations of x and y that present themselves as the outline of the discrete disc. The combination of a and b presents itself as c in order to highlight this fact and indeed show the resemblance of the outside of the discrete disc to the composition of these two Pythagorean triples. The solid square represents the centre of the discrete disc.

The initial inspiration for this work originated from the interference of waves that arose in the nodal formations on Chladni plates and indeed from instances of the wave equation and nodal patterns in different media. In this light the initial inspiration was carried through in the design of the messages that are passed and the way in which they are combined and used for computation. In this way the messages were designed to model waves, discretised in the form of words drawn from the alphabet Σ where here and in most of the rest of the work the input and output alphabets will, necessarily, be the same, $\Sigma = \Lambda$. Such combinations of messages were modelled in an attempt to generate a form of discretised wave equation complete with nodal patterns on the two dimensional plane in much the same way as the Chladni plates.

As such, messages, and the automata constructed to generate, pass and process such messages, are mostly used to simulate the transmission of a wave. Waves are discretised by taking equidistantly spaced values from time steps along a sine wave, although naturally such waves can be permuted to represent a cosine wave or indeed a wave which holds any value at time $t = 0$. The waves modelled may be taken to be of any amplitude and frequency with any resolution (number of values that represent the discretised wave) though naturally these will all affect the way in which they interact under the model and the resultant effects of such combinations. The discretisations are represented more generally as words, $W = w_0w_1w_2w_3...w_{n-1}$ where $|W| = n$ (indicating the size of the word), such that any two values which are the same are represented by the same symbol drawn from, $\Sigma = \Lambda$, the input/output alphabets for the automata.

Automata are constructed in such a way that the words that represent waves are combined and processed in a way which may model waves, that is in this sense the addition function which represents superposition in the classical sense for waves in a physical setting [30]. Although the classical model of waves uses the addition function for the composition of waves later other functions shall be examined for their properties in the resultant pattern formation in the medium. In the Broadcasting Automata model such a method of wave passing is model via the use of automata and their input and output alphabet. In the asynchronous alphabet, to represent a wave of four values, as seen in Figure 2.8 omitting fractional values (± 0.5), it is necessary to construct an automata that is able to accept the preceding value in the series, for example if the wave is represented in the following series of integer values, $0, 1, 0, -1$, and translated to a word such as, $W = w_0w_1w_2w_3$, then the automata must be able to construct upon receipt of the symbol, w_i, the proceeding symbol, w_{i+1}. In this case upon receipt of w_1, or the integer 1, the automaton must be able to accept the symbol and upon state transition emit the next message in word, W, that makes up the wave, in this case, 0. Now that the automaton has received the symbol, w_1, it must implement some method of combining this symbol with another symbol that represents superposition, in two dimensions a simple addition of the amplitude, here the combination of symbols from W. Such emissions of symbols are cyclical, just as a wave is, and as such in the case of $W = w_0 w_1 w_2 w_3$ the next symbol to be emitted after a receipt of w_3 is w_0. More generally for any word W of size $|W| = n$ the next symbol after w_i is $w_{(i+1) \mod n}$.

Definition 2.13. A discretised wave is a word, W, such that each symbol, $w_i \in W$, represents an amplitude, equidistant from the last, and such that within one wavelength for each equidistant division of amplitude, y, both positive and negative, all solutions for x (i.e $sinx = y$) are a new symbol in the word, W, where all symbols are ordered by their monotonically increasing, x, value. In this case time.

Such a method of combining symbols that represent fractions of waves may be implemented using states and additional output symbols that are relayed upon request, using the state request symbol, σ_{st}, so that neighbourhoods can be calculated, which shall be heavily used later, chapters three and four, in algorithmic constructions. As an example, upon receipt of w_3, after the receipt of w_1, the automaton makes the transition to a state, which could be a final state if the computation only requires the passing of two waves to construct information about the plane as it usual in later algorithms, that represents the addition of the two, in this case the addition of $w_1 = 1$ and $w_2 = -1$ which can be represented as a remapping to the state with output symbol w_0 or w_2 where as if the out put were the additive product of (w_1, w_1) then a new output symbol representing 2 is required. Such a Cartesian product of states can be represented as a new state for each possible combination or, as is the case for a simple addition, replaced with a symbol for the possible unordered pairs. In fact any number of methods for mapping the two symbols in to states, that have as their output the symbol for the combination of the pair, can be designed. In chapter three, information that can be gleaned from such mappings of combinations of states and what this infers about the history of the waves that they represent is more fully explored. Here, only an notion of how such constructions can be achieved is given.

Figure 2.8 gives an example of the digitisation of a wave showing where the equally spaced amplitudes occur on the sine wave. Here the wave is of amplitude one (and if time is to be considered and equal to one also then this would show a frequency of one hertz) and divided in to four equally spaced parts which give the following four values as time progresses positively from zero $0, 0.5, 1, 0.5, 0, -0.5, -1, -0.5$. In some cases it may be beneficial to substitute the real values that represent the wave with symbols over a finite alphabet. Indeed this representation may be more in line with the definition of broadcasting automata that has been presented by the model. For example the points $0, 0.5, 1, 0.5, 0, -0.5, -1, -0.5$ are now modelled as a word such that $W = w_0w_1w_2w_3w_4w_5w_6w_7w_8w_9$, where, , and where $w_i \in \Sigma = \Lambda$ for all $0 \leq i < |\Sigma|$, W is a word over Σ. Depictions of three successive steps of broadcast where waves are modelled as waves and in the asynchronous model for both $r^2 = 1$ and $r^2 = 2$ are given in Figure. 2.9.

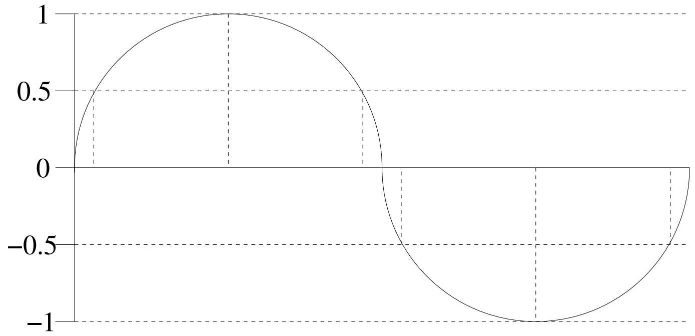


Figure 2.8: Depicting an analogue wave digitised by taking values from it that are equidistant in amplitude. Here, the x-axis depicts time where the y-axis depicts amplitude.

An example of the model as shall be used throughout the rest of the thesis is given in Figure 2.9. Here, the central point, which has been labelled, u_0, denotes in both cases, those that form square and diamond waves, is the initial transmitter, $A \in V_0$, and has been set accordingly at the initial state, shown here as the output symbol that would be generated, $\tau(q) \in \Lambda$ for $q \in Q$. All of the squares represent, at the central point within such a square, a single automata, A, at the point of its mapping within the configuration, where here $c(v) = u_0$ and

$v \in M$ is the point in the Euclidean space, (M, d). In the example on the left in Figure. 2.8 all states of the automata, $q \in Q$, have the same transmission radius such that, $\tau(q) = 1$. For all of the automata in the right example the transmission radius is, $\tau(q) = \sqrt{2}$ for $q \in Q$. These choices of radius generate equivalent neighbourhoods for transmission as the Moore and Von Neumann neighbourhoods. When the automata at the centre, here denoted with its state, u_0, here at some point $v \in M$, transmits this symbol to its neighbours, they receive the symbol u_0 and act accordingly, making the transition in to the state which has as its output symbol, u_1, such that for all automata within distance of the transmission radius, $d(v, v') \leq \tau(q) = 1$ for all $v' \in M$ and $q \in Q$ there is the transition, $\delta(q_0, u_0) = u_1$. The continuation of this process for all other automata generates the wave pattern that is shown on the left in Figure. 2.8 and with a replacement of all the states of the automata having a transmission radius of, $\tau(q) = \sqrt{2}$, the computation on the right can be formed. The cyclical nature can be represented here, assuming that there are only a total of four messages that can be passed which represents four points of a discretised wave, by having a transition such that, $\delta(q_0, u_3) = q_1$ where $\Delta(q_1) = u_0$.

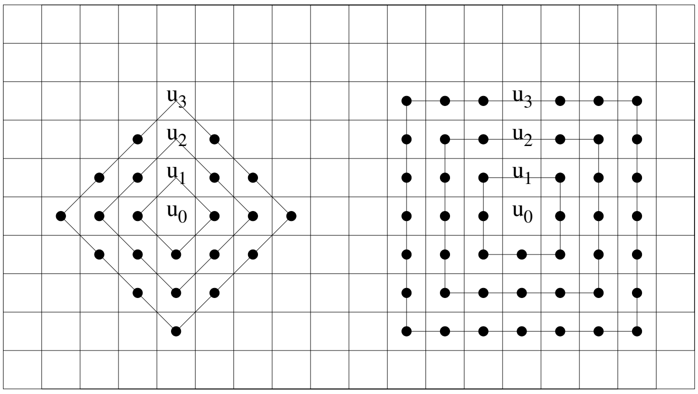


FIGURE 2.9: Wave propagation with the asynchronous model is shown on the square grid: Moore neighbourhood (left) and Von Neumann neighbourhood (right).

With this specific model it is still possible to design many primitives in a similar style as designed for the cellular automata model [65], including synchronisation procedures, finding the edge elements on a line, shortest branch of a tree, etc. These basic tools, based on ideas from cellular automata, will be utilised to illustrate more sophisticated algorithmic methods such as a solution to the Firing

Squad Synchronisation Problem [66] and locating the centre of discrete disc which will be presented later on in the thesis.

2.3 Neighbourhood and Broadcasting Sequences

The idea of propagating a pattern of square ($r^2 = 2$) or diamond waves ($r^2 = 1$), generated by repeated application of Moore or Von Neumann waves respectively, in dimension two are commonly known as Neighbourhood Sequences. Neighbourhood Sequences are an abstraction used to study certain discrete distance metrics that are generated upon the repeated application of certain neighbourhoods to a lattice for example the Moore neighbourhood to a square lattice. As previously discussed such neighbourhoods as von Neumann and Moore are encompassed by the more general model used here whereby the transmission radius is varied on a square lattice the first two of such transmission radii to produce distinct objects equating to those of the von Neumann and Moore neighbourhoods.

Definitions and notation concerning neighbourhood sequences as considered in [35], and many other works, are now given with deviations considered in this thesis highlighted after the more traditional treatment here.

Let $p \in \mathbb{Z}^n$ where $n \in \mathbb{N}$ and such that the ith coordinate of p is given by $Pr_i(p)$ for $1 \leq i \leq n$.

Definition 2.14. For $M \in \mathbb{Z}$ where $0 \leq M \leq n$ the points $p, q \in \mathbb{Z}^n$ are $M - Neighbours$ when the following two conditions hold:

- $|Pr_i(p) - Pr_i(q)| \leq 1$ for $(1 \leq i \leq n)$
- $\sum_{i=1}^{n} |Pr_i(p) - Pr_i(q)| \leq M$

An n-dimensional neighbourhood sequence is denoted $\mathcal{A} = (a(i))_{i=1}^{\infty}$, $\forall i \in \mathbb{N}$ where $a(i) \in 1, ..., n$ denotes an $M - neighbourhood$ by its value of M such that for $a(1)$ denotes that the next neighbourhood set of points in the sequence is those that differ by at most one coordinate as given by Definition. 2.14 where $M = 1$ in this case the Von Neumann neighbourhood. If $\mathcal{A}$ is periodic then $\exists l \in \mathbb{N}$, $a(i+l) = a(i)$ $(i \in \mathbb{N})$. Such periodic sequences are given as $\mathcal{A} = (a(1), a(2), ..., a(l))$.

Definition 2.15. The $\mathcal{A}$-distance, $d(p, q; \mathcal{A})$, of p and q is the length of the shortest A-path(s) between them.

As the spreading of such neighbourhoods is translation invariant only an initial point of the origin need be considered w.l.o.g.

Definition 2.16. The region occupied after k applications of the neighbourhood sequence $\mathcal{A}$ is denoted as $\mathcal{A}_k = \{p \in \mathbb{Z}^n : d(0, p; \mathcal{A}) \leq k\}$ for $k \in \mathbb{N}$.

Also, let $H(\mathcal{A}_k)$ be the convex hull, given in Definition. 2.17, of $\mathcal{A}_k$ in $\mathbb{Z}^n$.

Definition 2.17. A convex hull is the smallest set of points that form a convex set as given in Definition. 2.18.

Definition 2.18. Convex set. A set $C \subset R^d$ is convex if for every two points $x, y \in C$ the whole segment xy is also contained in C. In other words, for every $t \in [0, 1]$, the point $tx + (1 - t)y$ belongs to C. [67]

Discrete discs are formed by the Broadcasting Automata as they are arranged on the plane at integral Cartesian coordinates, $(\mathbb{Z} \times \mathbb{Z}) \in M$, and as such a broadcast to all automata in range, for point $v \in M$, $\tau(c(v)) = r$ will cause a change in state to all those automata that form a discrete disc of radius r from the point v. In the broadcasting automata model it is assumed that there is no restriction on the radius of transmission where it is considered that as the Von Neumann and Moore neighbourhoods can be described as the first two radii in the set of distinct discrete discs, $r^2 = 1$ and $r^2 = 2$. This means that discrete discs are quite a natural extension to the basic notion of neighbourhood sequences merely relaxing the constraints on the initial definition of $M - neighbours$ in the following way.

Definition 2.19. Two points $p, q \in \mathbb{Z}^n$ are $r - neighbours$ if the Euclidean distance, $d(p, q) = \sqrt{\sum_{i=0}^{n}(q_i - p_i)^2}$, is less than some r, used to denote the radius of a circle, such that $d(p, q) \leq r$.

Utilising the framework supplied by the work done on neighbourhood sequences it is no possible to outline **Broadcasting sequences** which denote any sequence of radii, r, such that $R = (r_1, r_2, ..., r_l)$. Labelling those points that are reachable by some application, R_k, of the neighbourhood sequence is another extension to the notation. All points such that $p \in R_1$ are labelled 0, all points $p \in R_2 \backslash R_1$ are labelled as 1. More generally labels will be assigned b where $k \equiv b \mod m$ and $m \in \mathbb{N}$. The work conducted in this thesis will also be restricted to the study of $\mathbb{Z}^2$.

Many of the questions asked and, indeed, answered, in neighbourhood sequences, shall be useful in the exposition of broadcasting automata. These include whether a certain sequence of neighbourhoods are metrical or not [44]. Indeed when using a general form of broadcasting automata whereby alternation of the broadcasting radius is allowed at each step as suggested here. Considering that all notions of the construction of algorithms presented here in the broadcasting automata model rely on the distance from the transmitter to the node, being able to assure that this distance will be consistent for all automata is essential. Also studied in neighbourhood sequences is the possible resultant shapes which are categorised and examined for their various properties such as their use in approximation of euclidean distances where the isoperimetric ratio is used to compare the sequences based on the shapes that they form on the plane [38]. This same method is used to again show and estimation for euclidean distance as well as estimations for certain L_p distances, here the astroid which will be presented in chapter six. The characterisation of the shapes generated by these neighbourhood sequences yield results about the shapes of compositions of such discrete discs on the plane and the partitions that such compositions form.

Chapter 3

Discrete Interference of Waves

The concepts of broadcasting automata and discrete informational waves introduced in Chapter 2 have much potential for the organisation of distributed computation, pattern formation and the design of new principles for defining metrics in the discrete digital environment. This chapter will start on the exploration of novel techniques and methods for pattern formation by coupling the already well known mechanisms in the automata model with several notions present in natural phenomena such as waves, superposition, nodal patterns and normal modes.

In physics, a standing wave is a wave that remains in a constant position. This phenomenon can occur in a stationary medium as a result of interference between two waves travelling in opposite directions. A standing wave in a transmission line is a wave in which the distribution of current value is formed by the superposition of two waves propagating in opposite directions. The effect is a series of nodes (zero displacement) and anti-nodes (maximum displacement) at fixed points along the transmission line. In between these lines there are points (or regions) of oscillation which are bounded by some minimum or maximum. With standing waves in a two dimensional environment the nodes become nodal lines, lines on the surface at which there is no movement, that separate regions vibrating with opposite phase [30].

All of the above affects, already thoroughly explored in physics, can be observed as components of the *Broadcasting Automata* model on a square lattice. Moreover the patterns formed by standing waves can be made more complex than the original since, even with the finite memory each automata may recognise not only a constant value but also any finite periodic sequence of values. The characterisation of discrete nodal patterns and novel techniques for partitioning are the main results of this chapter and also constitute important tools for organising distributed computations based on the broadcasting primitives formalised in the preceding chapter.

3.1 Modelling Waves Using Broadcast Automata

This chapter will define the discretisation of waves and explore the results of such a system on mediums in $\mathbb{Z}^1$ and $\mathbb{Z}^2$ using a transmission radius restricted to $r \in \{1, \sqrt{2}\}$ for $\tau(q) = r$ where $q \in Q$ representing the classical Moore or the von Neumann neighbourhoods and their applications in $\mathbb{Z}^1$ and $\mathbb{Z}^2$ which will be discussed later. The underlying positioning of the automata on the plane correlates with the square lattice. This is simplified for the emulation of discrete waves in a rigid medium presented here and as such the underlying positioning is given as each automata is at some Cartesian coordinate where all coordinates are integers. The wave is discretised as previously discussed with equidistantly spaced amplitude values taken from the appropriate amplitude and frequency of sine wave which are represented as words. The automata are simplified with states constructed to allow the functional composition of words. Here that function will mimic the composition found by waves in a physical environment, addition, as given by the superposition of waves.

More formally this leads to the model of Broadcasting Automata where (M, d) is $M = \mathbb{Z} \times \mathbb{Z}$ and where for $\rho = (\rho_0, rho_1)$ and $\rho' = (\rho'_0, \rho'_1)$ the distance function is $d(\rho, \rho') = \sqrt{(\rho_0 - \rho'_0)^2 + (\rho_0 - \rho'_0)^2}$, the Euclidean distance function for $\rho, \rho' \in M$. A more formal notion of the workings of the automata that construct the nodal patterns will be introduced later when a firm notion of what nodal is has been introduced and the ways in which it can be computed in a simplified way have been solidified.

As previously stated the discrete waves are modelled as words, W, drawn from the symbols that make up the input and output symbols of the automata, Σ, Λ, respectively and are formally constructed in Definition. 2.13. The symbols will represent equidistantly space amplitudes of a sine wave and each symbol in the wave will be referenced by its index in the word W such that $w_i \in W$. The waves that are produced by sources or transmitters, the initial active automata defined as $A \in V_0$, may be different for each transmitter and correspond to any wave of any amplitude, frequency combination. Just as the combinations of amplitude from the waves present create the amplitude observed in the medium so the words combined by the automata mimic such physical properties, combining the models of amplitudes, words, via states, which represent the superposition of such amplitudes via some function and through the transmission sequences which model the motion of waves through physical media. In this way the regions of the medium that oscillate through plus and minus some bounds are represented by the same oscillations of combinations of symbols over time. Such combinations of symbols are described as **nodal patterns**, differing from common physical notation employed these nodal patterns are not simply nodal points of zero motion but any set of values over time that are equivalent.

Nodal patterns will here be explained in the simpler model of a single dimension in which the two transmission radii produce the same underlying graph of transmission. The following explanation shall also use the more intuitive notion of simultaneous transmission through the medium which as shall be shown later is not possible in the model. However, it is possible to show that this more intuitive model of waves as well as the synchronous and asynchronous models are equivalent which shall be done once the correlation between distance and time is proved. The one dimensional explanation presented here also expands in to a two dimensional explanation in much the same way solutions to the two dimensional wave equation are expansions of the one dimensional wave equation. Indeed the following exposition is a simplification through abstraction of the model which, as only one construction of the automata is used, and explained later when it becomes apparent that such automata are easily reducible to a few states, the model need not reason with such a level of detail. The following exposes this.

Two transmitters, $T_1 \in V_0$ and $T_2 \in V_0$, are placed on a one dimensional line such that each automata is placed at coordinate $x \in \mathbb{Z}$ and there is distance d between the two automata. When restricted to one dimension the transmission radii, $r \in \{1, \sqrt{2}\}$, differs for neither choice, each radius is simply able to transmit to its neighbours at $x+1$ and $x-1$ from an automata at coordinate x. Each initial automata, T_1, T_2, broadcasts words u^* and v^* respectively, where $*$ is the Kleene star, $u, v \in \Sigma \subseteq \Lambda$ where Λ may be bigger as, under composition, the words that form the output wave may not be closed (i.e under composition two waves of the form $0, 1, 0, -1$ contain the element $1 + 1 = 2$) and where $|u| = |v| = s$ is the length of the word. The broadcasting of symbols begins at transmission points with symbols broadcast away from the source at each time step, cycling through all symbols in the word such that the front of the 'wave' is always $v_0 \in v$ or $u_0 \in u$. We now have two infinite words $(u)^*$ and $(v)^*$, the reverse of v, that are broadcast towards the opposing transmitter, from one automata to the next, on every time step.

It is possible to see a correlation between the time steps, t, and the value of the symbol for each coordinate x on the line. Assume that transmitter T_1 is placed at the origin, i.e $x = 0$ for T_1, then let $N(x_T, x, t)$ be the function that associates the automaton at coordinate x and time t with an index i of a symbol in the word what represents the wave, $w_i \in W$ and x_T be the coordinate of the transmitter, which in this case will be 0. In this instance the mapping can be formed such that $N(x_T, x, t) = (x, w_{(t-(x-x_T)) \pmod{|W|}})$ assuming that the transmitter is at the origin and that any mapping $N(x_T, x, t) = (x, w_{(t-(x-x_T)) \pmod{|W|}})$ where $(t-(x-x_T)) < 0$ refers to the quiescent state, q_0, as given in the definition of the Moore machine. Such a mapping associates an automaton at coordinate, $x \in \mathbb{Z}$, with a symbol, $w_i \in W$, for a time, t, the pair, (x, w_i). This correlates to a higher level abstraction of the configuration function, given in Definition. 2.6, which associates a point with

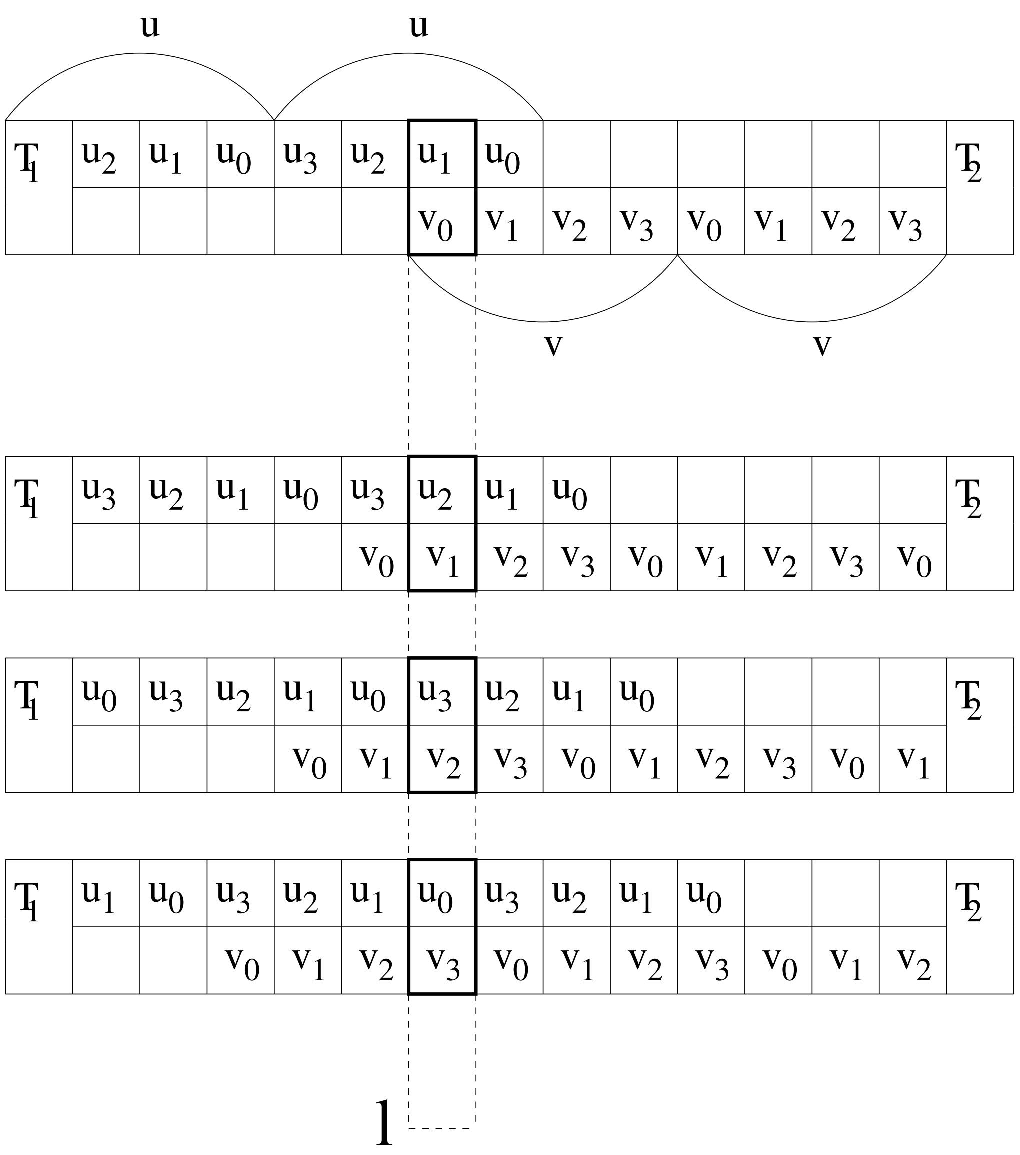

u u
T1 u2 u1 u0 u3 u2 u1 u0 T2
v0 v1 v2 v3 v0 v1 v2 v3
v v
T1 u3 u2 u1 u0 u3 u2 u1 u0 T2
v0 v1 v2 v3 v0 v1 v2 v3 v0
T1 u0 u3 u2 u1 u0 u3 u2 u1 u0 T2
v0 v1 v2 v3 v0 v1 v2 v3 v0 v1
T1 u1 u0 u3 u2 u1 u0 u3 u2 u1 u0 T2
v0 v1 v2 v3 v0 v1 v2 v3 v0 v1 v2
1

Figure 3.1: Shows two transmitters T_1 and T_2 broadcasting words u and v respectively where time, t, is shown as motion down the page.

a state, $c(x)$, for $x \in M$ where in this case $M = \mathbb{Z}$ and time represents a number of transitions via the global transition function, $\mathcal{G}$ as given in Definition. 2.8.

If the transmitter is not at the origin, which is naturally the case for the second transmitter T_2 then again, $N(x_T, x, t) = (x, w_{(t-(x-x_T)) \pmod{|W|}})$, where x_T is the coordinate for the location of the transmitter in this case T_2. This holds here because the distance that can be travelled by each transmission for each time step is one coordinate in either direction as given by the transmission radii, r, though naturally this can be varied along with the dimension. Using the Euclidean distance function it is possible to say that all of those points within distance, d, that are less than or equal to the time step, t, have received a symbol, $w_i \in W$.

For simplicity of analysis and intuition, highlighting the correlation with the natural phenomena that inspired broadcasting automata, it is assumed that the two waves may move in opposite directions at the same time however, later on in the chapter, it shall be shown that this is not the case for this model but that these differences are reconcilable and reduce to the same representation of patterns so that, where convenient, the reader may consider this to be a reasonable model for thought experiment.

Once an automata has received as many symbols as there are transmitters, in this case two, it is able to decide to which of the nodal patterns it belongs. In this case any automata on the plane at a point $\rho \in \mathbb{Z}$ which has received two symbols, one from the word u and one from the word v, forms a pair $p_{i,j} = (u_i, v_j)$ where such pairs are not ordered. Once a point ρ is in such a configuration, it is possible to define a sequence of pairs $\rho_{(i+t) \pmod s),(j+t) \pmod s)}$ contained in l over discrete time, $t = t', t'+1...t'+s$. It is easy to see that such sequence will be periodic with a period less or equal to s, where $|u| = |v| = s$, and it represents a history of symbol pairs from u and v which are meeting at the point l over time. It is now possible to give a more formal definition of a **nodal pattern**.

Definition 3.1. A **nodal pattern**, P, of a point, $\rho \in \mathbb{Z}$, is a finite subsequence of s pairs $p_{(i+t) \mod s,(j+t) \mod s} \in P$ over some $t = t'..t'+s$ where s is the length of the words that are used to generate the pairs and t is sufficient such that for $N(x_T, x, t) = (x, w_{(t-(x-x_T)) \pmod{|W|}})$ there is $(t-(x-x_T)) > 0$ for all transmitters, x_T.

It is also assumed that nodal patterns are equivalent up to a cyclic shift, so it will not be dependent on the initial time t'.

Definition 3.2. Two nodal patterns, P and Q are said to be equivalent if they can be made to be the same using some cyclic shift of the pairs. Otherwise they are said to be distinct.

For example, a wave with a periodicity of four is taken such that it has the form, $0, 1, 0, -1$, and may be formed as the word, $W = w_0w_1w_2w_3$ in accordance with Definition. 2.13. Two formations of nodal patterns are constructed for two different points where Figure. 3.1 shows the construction of a series of pairs over time, denoted by the column l, for the words, u and v. Here, it is possible to use the equation, $N(x_T, x, t) = (x, w_{(t-(x-x_T)) \pmod{|W|}})$, where the point, $x \in \mathbb{Z}$, is equivalent to the point, $\rho \in \mathbb{Z}$, such that the two tuples formed, from both transmitters, for Figure. 3.1, T_1 and T_2, generate the pairings. The set of pairs that are formed

$$P = ((N(x_{T_1}, \rho, t), N(x_{T_2}, \rho, t)), (N(x_{T_1}, \rho, t+1), N(x_{T_2}, \rho, t+1))(N(x_{T_1}, \rho, t+2),$$

$$N(x_{T_2}, \rho, t+2))(N(x_{T_1}, \rho, t+3), N(x_{T_2}, \rho, t+3)))$$

for point $\rho \in \mathbb{Z}$, make up the nodal pattern, P, and similar is done for Q. This yields for time, t, and points, ρ, ρ', the nodal patterns, P and Q as follows where equivalence can be seen of such points via a cyclic shift:

$$P = ((1, 0), (0, -1), (-1, 0), (0, 1))$$

becomes equivalent to

$$Q = ((0, 1), (1, 0), (0, -1), (-1, 0))$$

when Q is cyclically shifted by 3 to the right such that it becomes

$$Q = P = ((1, 0), (0, -1), (-1, 0), (0, 1)) \ .$$

Nodal patterns are exemplified in Figure. 3.1 where $s = 4$, $u = u_0, u_1, u_2, u_3$ and $v = v_0, v_1, v_2, v_3$. At point l the pairs $((u_1, v_0), (u_2, v_1), (u_3, v_2), (u_0, v_3))$ form a nodal pattern corresponding to $p_{(1+t) \bmod s, (0+t) \bmod s}$ over time t. Here the waveform is represented by the words u and v and as such it can be said that, $0, 1, 0, -1$ can be formed in to the words, u, v such that, $u_0u_1u_2u_3 = v_0v_1v_2v_3$.

Indeed such reasoning may be applied to any of the waveforms that are generated by digitising a wave form of any amplitude for example, $0, 1, 2, 1, 0, -1, -2, -1$ such that $0, 1, 2, 1, 0, -1, -2, -1$ gives words, u, v such that, $u_0u_1u_2u_3u_4u_5u_6u_7 = v_0v_1v_2v_3v_4v_5v_6v_7$.

Looking at all of the resultant patterns that may occur from intersection on the plane and assuming that the method of composition is the same as that found in physics, addition, then the following possible nodes are constructed:

$$\begin{gathered}((1,1),(0,0),(-1,-1),(0,0)) \\ ((1,0),(0,-1),(-1,0),(0,1)) \\ ((1,-1),(0,0),(-1,1),(0,0)) \\ ((1,0),(0,1),(-1,0),(0,-1))\end{gathered}$$

which, under composition, such that for all pairs the addition function is applied , yields the following:

$$\begin{gathered}(2,0,-2,0) \\ (1,-1,-1,1) \\ (0,0,0,0) \\ (1,1,-1,-1)\ .\end{gathered}$$

From the above derivations it can be seen that there will only be a finite number of values which every automata will find itself oscillating through which directly parallels what would be found in a physical medium. The traditional nodal pattern $(0, 0, 0, 0)$ can be found, where there is no motion, but also in this model it is desirable to extend any set of periodic oscillations to the definition of nodal pattern such that as long as it is distinct from another pattern, which, along with the whole body of knowledge on this subject, shall be firmly defined in the next section, shall constitute a distinct nodal pattern.

As previously discussed the nodal patterns considered here are for words of the same length which are identical and where the function for aggregation is addition which mimics that of superposition in nature. It shall be seen that this it is not necessary for the function of aggregation to be addition in the Chapter 6 which examines alternatives at great length. These are directly equivalent to the number of distinct nodes that are possible to form when all transmitters are transmitting the same wave, i.e $W = 0, 1, 0, -1$ for the wave of amplitude one.

It is also noted that the infinite repetition of the words upon broadcast generates a cyclical nature for the discrete wave that closely correlates with the cyclicality of waves in nature. Given the finite and cyclical nature of u and v as they are broadcast throughout the network it is possible to represent each of the nodal patterns as a single pair of indices, when only considering two transmitters as is done here, which helps to greatly simplify notation whilst conveying the same information. As the notation will always assume that the first element of the pair will be of index 0 only the second index will be used to signify the unique pairings that are possible for the two words, u and v. This is equivalent to the absolute value of the two indices where

$$|(i+t) \pmod s - (j+t) \pmod s| = |i-j|$$

and as such represents the constant distance between the two pairs over time allowing notation to be reduced to $P_{|i-j|}$.

Lemma 3.3. *For any two finite words there are s possible labels, $P_0, P_1, \ldots, P_{s-1}$, one for each of the possible pairings, though they are not all distinct as has been shown.*

Proof. The proof is derived from the equation, $|(i+t) \pmod s - (j+t) \pmod s| = |i-j|$, where any constant shift between the two indices, i and j, representing motion over time, time is cancelled when calculating the distance between the values of the intersection in the definition for a nodal pattern, Definition. 3.1. □

Such indices, that subscript each P, are now defined as the **nodal index** of the automaton, which signify the distinct pairing observed by the automata. The following proposition looks at counting the number exact number of distinct pairings that are generated using this definition of a nodal point.

Proposition 3.4. *Given a non-periodic word u where $|u| = s$. The number of possible distinct nodal patterns generated by broadcasting words u and v, where $v = u$, is $s/2 + 1$ if s is even and $(s + 1)/2$ if s is odd. Further, the patterns P_k and $P_{s-k \pmod s}$ are equivalent up to cyclic shift.*

Proof. We can represent the pairings that form nodal patterns by the permutation group that contains the set of permutations σ for which the permutations will be generated by the function $\sigma_i(u_j) = v_{(j+i) \pmod s}$ where $0 \leq i, j < s$.

For example one element of the group such that σ_1 for some word of size $s = 4$ would be

$$\sigma_1 = \begin{pmatrix} u_0 & u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 & v_0 \end{pmatrix} .$$

As the symbols of each word, u and v, are unordered pairs it is possible to equate each permutation with its inversion, the permutation formed by switching all u for all v. The inverse of each permutation is exactly that permutation formed by its inversion, swapping all symbols of u for those of v, which can be noted from the fact that $\sigma_i^{-1}\sigma_i = I$ where $\sigma_i(u_j) = v_{(j+i) \pmod s}$ and the inverse must now be $\sigma_i^{-1}(v_{j+i}) = u_j$ clearly the inversion of the u and v. Counting the number of permutations with inverses that are also distinct members of the group, that is the inverse is not itself where $\sigma_i\sigma_i = I$, amounts to all those permutations where both of the values of σ_i and σ_j sum to s, naturally, as these are all of the permutations that compose to the identity and also equivalent to those elements P_k and $P_{s-k \pmod s}$. This means that if s is odd then half of the set of permutations must have an equivalent permutation within the same set except for the identity which is of course its own inverse. If the set of permutations is even then there is a central element $\sigma_{s/2}$ which is also its own inverse adding another distinct element. □

Illustration of all of the possible combinations of nodal patterns can be seen in Figure. 3.2 where the structural equivalence through commutativity can be seen by the symmetry in the bipartite graph between words u and v. Those that are equivalent have been highlighted by using the same colour to represent the enclosing box.

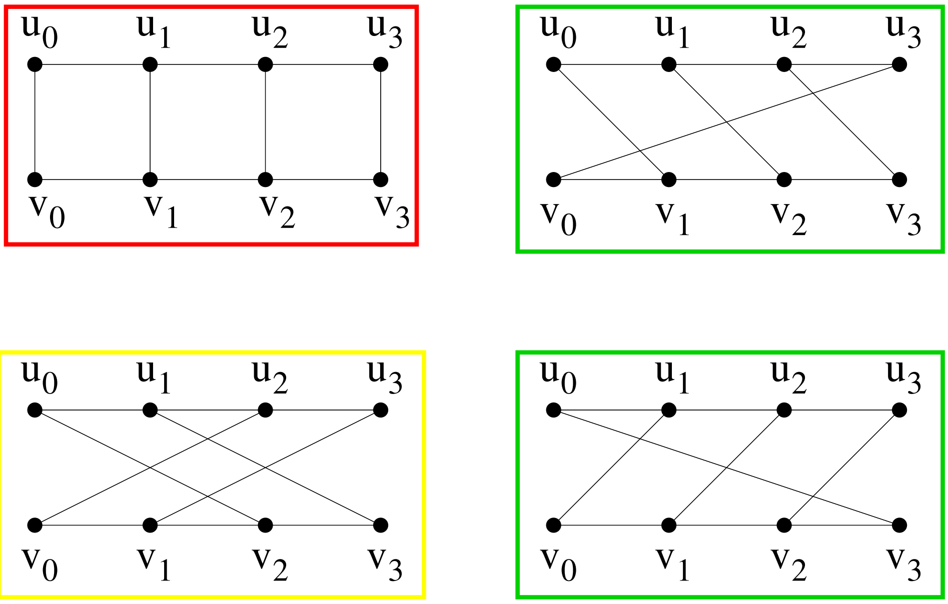


FIGURE 3.2: The structure of the nodal patterns constructed from words of length $s = 4$ exhibited as graphs where coloured boxes indicate those which are equivalent where $u = v$ and the function of combination is commutative.

The following example again illustrates the number of combinations of pairs that can be generated this time explicitly showing the tuples.

Example 3.1. *Given a sequence* $u = v = 1, 0, -1, 0$ *and so* $|u| = |v| = s = 4$ *it can be shown that there are* $(s/2) + 1 = (4/2) + 1 = 3$ *distinct nodal points. Enumerating all possible patterns (modulo removed for clarity) and where t is over time.*

$$P_0 = ((1,1), (0,0), (-1,-1), (0,0))$$
$$P_1 = ((1,0), (0,-1), (-1,0), (0,1))$$
$$P_2 = ((1,-1), (0,0), (-1,1), (0,0))$$

$$P_3 = ((1,0),(0,1),(-1,0),(0,-1))$$

Patterns P_1 and P_3 are the same assuming that the order of the symbols is not relevant i.e. $(-1,0) = (0,-1)$*, as* $P_1 = p_{(0+t),(1+t)} = ((1,0),(0,-1),(-1,0),(0,1))$ *becomes equivalent to* $P_3 = p_{(0+t+3),(3+t+3)} = ((0,1),(-1,0),(0,-1),(1,0))$ *when* P_3 *is cyclically shifted by* 3*.*

Proposition 3.5. *It is possible to construct the automaton which is able to elect itself, such that the output symbol of the automatons final state, F, corresponds to one of the distinct members of the set of nodal patterns, P, by receipt of only one message per initial transmitter which are automata in V_0.*

Proof. Such an automaton is derived from the result in Lemma. 3.3 which shows that only two symbols from the alphabet are required to define a nodal pattern as being a member of the set, P. As such it is possible to construct the automaton that takes two symbols, u_i and v_j, and based on their, i and j values elects the member of the set P that the nodal pattern belongs to by a simple process of enumerating the combinations in states. This implies that for ever set of symbols such that $|i-j|$ is an element of P, such that, $P_{|i-j|} \in P$, then there is a transition for symbols that match $|i-j|$ to a final state that represents the nodal pattern, $P_{|i-j|}$. □

It is noted that this method of constructing nodes is not the only possibility that may be considered. Indeed there are many permutations of this problem of counting distinctness. The problem may be extended to include words of differing sizes, differing symbols and they may be composed with any number of functions. There are more general solutions to a slight extension of this problem for n differing transmitters which may be solved using an encoding in the pólya enumeration theorem and utilising partitioning. Also a trivial upper bound is known to be equivalent to the size of the words, considered to be different, and based upon the non-commutativity of the function of composition.

Much work within this area has been done regarding estimating the size of distinct sum sets, where bounds on the number of distinct combinations of sets of number over the composition using addition are required, which illustrates the difficulty of providing the full picture with respects to these questions. Results are already well noted for general sum sets where combinations are considered only over addition [68] but not without the added constraint of permutable vectors as has been addressed here.

3.1.1 Distance Functions and Model Equivalence in $\mathbb{Z}^2$

Standing waves and nodal patterns may be formed when u^*, where $|u| = s$, is transmitted from one source T_0 and is retransmitted from another source T_1 after reaching it. Any point in $\mathbb{Z}^2$ having both signals from T_0 and T_1 will have identified its nodal pattern as the 'difference' between them as addressed with concerns to the standard two transmitter model.

Nodal patterns depends upon the distance and the time from the initial transmitter and transmission respectively. Let us assume that the symbol w_i would have been received by the vertex at coordinate x from the transmitter at coordinate x_T at time t. In order to refer to the index i for w_i the function $N(x_T, x, t) = (x, w_{(t-(x-x_T)) \pmod{|W|}})$ is defined. This forms the very intuitive notion of waves that may pass at over each other simultaneously and is directly comparable to the notion that is seen in the physical world. However such a representation is not possible in broadcasting automata model. In this model due to the message passing primitive for both the synchronous and the asynchronous models whereby the automata sends a message and then refuses all other messages for a single round so as not to receive the message it has just sent. This affords broadcasting automata a simple way to send messages such that they are broadcast away from the transmitter which in turn generates the wave formations. As such the construction of nodal patterns on the plane is examined here with respects to the model for broadcasting automata that had been described. It shall be seen later that the nodal patterns, used for the construction of algorithms, are structurally equivalent regardless of the method of propagation used. This section aims to bring together and show the clear equivalence of both models, by showing propagation equivalence, and dimension equivalence. Both of these goals may be made possible by discussing the distance functions that represent the transmission of Moore and Von Neumann waves by broadcasting automata.

As already addressed in waves for one dimension there is no difference to the resulting propagation when using either a Moore or Von Neumann transmission pattern. However, upon moving the waves to two dimensions, $\mathbb{Z}^2$, it is important to define the distance between the points in terms of (square or diamond) wave propagation for correct calculation of the nodal patterns, as it is not the same as in Cartesian geometry. The following definitions correspond to the well known L^p spaces, specifically, L^1, representing the Von Neumann neighbourhood and, L^∞, representing the Moore Neighbourhood where, L^2, is the standard Euclidean metric [69].

Definition 3.6. Square waves adhere to the distance function $d_\square$ for distances from p to q where $d_\square : \mathbb{Z}^n \times \mathbb{Z}^n \to \mathbb{Z}$ and is defined by the correspondence $d_\square((p_1, \ldots, p_n), (q_1, \ldots, q_n)) = \max_{1 \leq i \leq n}\{|p_i - q_i|\}, (p_1, \ldots, p_n), (q_1, \ldots, q_n) \in \mathbb{Z}^n$.

Definition 3.7. Diamond waves adhere to the distance function $d_\Diamond$ for distances from p to q where $d_\Diamond : \mathbb{Z}^n \times \mathbb{Z}^n \to \mathbb{Z}$ and is defined by the correspondence $d_\Diamond((p_1, p_2, ..., p_n), (q_1, q_2, ..., q_n)) = \sum_{i=1}^{n} |p_i - q_i|, (p_1, p_2, ..., p_n), (q_1, q_2, ..., q_n) \in \mathbb{Z}^n$.

Informally speaking, the distance functions defined above provide the time taken for the diamond/square wave front to reach an automata at coordinate $(p_1, p_2, ..., p_n)$ from a transmitter at $(q_1, q_2, ..., q_n)$.

Let us consider two ways of computing nodal patterns for this process in terms of synchronous and asynchronous cases with transmitters T_0 and T_1. In both cases we aim to get the same distribution of nodal patterns on $\mathbb{Z}^2$ without the need to continuously propagate values from the sources. It is also with this method that it will finally become apparent that all of the preceding intuition regarding these calculations, in which the waves are able to pass over each other much like occurrences in physical media, coincides with the reality of the calculation that happens within the model. The local distribution of these patterns will allow us later to locate points in a non-oriented environment.

In order to illustrate how the two different models, synchronous and asynchronous, can be reconciled with the more intuitive notion of how the waves might move in the plane the introduction of the following definition of a nodal pattern for a point on the plane will prove insightful.

Definition 3.8. An index i of a nodal pattern P_i, resulting from the transmission of two waves, in a point $\rho \in \mathbb{Z}^2$ is defined as

$$i_\rho = |d(T_0, \rho) - (d(T_0, T_1) + d(T_1, \rho))| \pmod s$$

where d is understood as $d_\Box$ or $d_\Diamond$ for square or diamond wave respectively.

This definition illustrates all that is needed to know about the way in which these nodal patterns are formed on the plane. It is the distance, which is, ultimately, equatable to time, as each automata takes a constant amount of time to process and pass the message, that is between the time taken for the first message, from the first transmitter, to reach the automata and the time taken for that message to reach the second transmitter, thus activating it, and for that second transmitters first message to reach the automata which solidifies the nodal pattern after the receipt of two messages. The following diagram depicts this labelling of the automata which is seen in Figure. 3.3.

Synchronous model with a single message. In case of transmitting a single message, waves here are activation waves, which the automata use to start internal clocks. Transmissions begin from T_0 where the activation wave propagates through the use of a square (or diamond) wave arriving at T_1 which is activated when reached by the first wave after a constant delay s. Any point that has received the first signal will start its internal clock (which counts modulo s), and then after receipt of the second signal the clock is stopped and the value of clocks

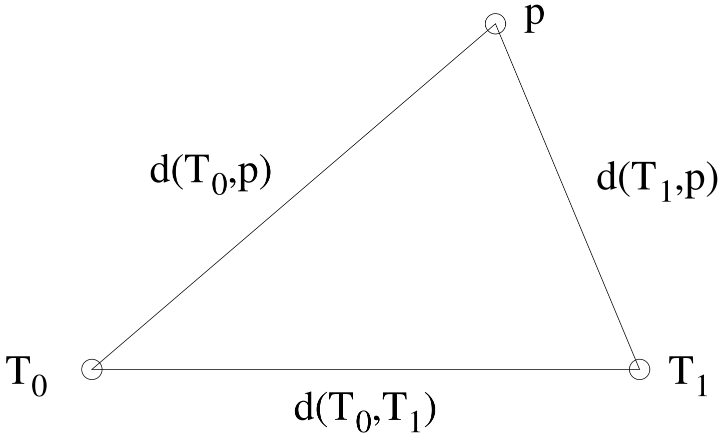


Figure 3.3: A depiction of the labelling of automata on the plane. Each of the distances are computed with the appropriate distance function for the shape of the broadcast, here, square or diamond. As the nodal pattern is decided by the distance between the first transmission to be received by p and the second the distances must be calculated as shown and the absolute difference found in order to label p.

corresponds to the index of the nodal pattern for this point which is $|(d(T_0, \rho) \pmod s) - ((d(T_0, T_1) \pmod s) + (d(T_1, \rho) \pmod s))|$.

Asynchronous model with multiple messages. In the case of the asynchronous model, the same distribution of nodal patterns can be simulated by sending a wave from T_0 where, on the wave front, every point that receives a symbol u_i immediately transmits the symbol $u_{(i+1) \pmod s}$. The synchronisation of wave propagation is achieved by assuming that every transmission takes the same constant time. Then transmitter T_1 operates in the same way once reached by u_i, transmitting the next symbol corresponding to u_{i+1} but using a different alphabet $\{v_1, ..., v_s\}$ to avoid problems whereby transmitting in the same alphabet could have a blocking effect on the wave. Each node should now contain a pair of symbols $(u_{i'}, v_{i''})$ which is enough to define the pattern $P_{|i'-i''|}$, where $i' = d(T_0, \rho) \pmod s$ and $i'' = (d(T_0, T_1) + d(T_1, \rho)) \pmod s$.

In the next theorem the properties of nodal pattern distribution is shown resulting in a new approach for partitioning $\mathbb{Z}^2$ via non-oriented transmissions and is one of the core tools for the geometric algorithms discussed in this thesis.

3.2 Nodal Patterns of Discrete Interference in $\mathbb{Z}^2$

This section details those nodal patterns that may be formed by the placement of two transmitters on the plane or, in the case of those patterns constructed from a single point it may be considered that one transmitter sends first one transmission then the next. These patterns will be essential in the construction of algorithms to locate the centre of a circle and can be considered to be central to the use of broadcasting automata to construct a whole host of algorithms and as such these patterns may be considered to be the basic building blocks of algorithms in the broadcasting automata model.

3.2.1 Nodal Patterns Resultant from Two Sources

A complex system of patterning can be achieved with the basic primitives that have been laid out so far. A characterisation of this patterning and a proof of its existence and stability are given here. Such reasoning is given for the diamond wave, however, with some basic substitution of distance function such patterning can also be derived for the square waves. It can be observed from Figure. 3.4 the direction of motion of the waves that are generated by repeated broadcast of the Von Neumann waves. The direction of motion of the wave describes the notion that, within the sections contained within black borders, a directional arrow indicates there are a series of lines, orthogonal to the arrow, that make up that section. Two arrows, one black, one white, indicate that in this sector there are two waves with the direction of motion as given by the arrow. The direction of the arrows give a good intuitive notion of how distinct nodal patterns are formed on the plane, both how and where. It can be seen from the combinations of wave direction, orthogonal, towards each other and in the same direction of motion, and the resultant patterns that form on the plane that there are three distinct patterns. Such patterns are given in the following way:

$\{D_9\}$ — has two nodal patterns,

$\{D_1, D_3, D_5, D_7\}$ — has one nodal pattern,

$\{D_2, D_4, D_6, D_8\}$ — has three nodal patterns:

The sectors containing the sets are given rigorous algebraic boundaries thus, which can be seen pictorially in Figure. 3.4.

$$D_1 = \{(x, y)|y \geq -x + (k_1 + j_1), y \geq x + (k_0 - j_0)\}$$
$$D_2 = \{(x, y)|y \geq -x + (k_1 + j_1), y \leq x + (k_0 - j_0), y \geq x + (k_1 - j_1)\}$$
$$D_3 = \{(x, y)|y \leq x + (k_2 - j_2), y \geq -x + (k_2 + j_2)\}$$
$$D_4 = \{(x, y)|y \leq -x + (k_1 + j_1), y \leq x + (k_0 - j_0), y \geq x + (k_0 + j_0)\}$$
$$D_5 = \{(x, y)|y \leq -x + (k_0 + j_0), y \leq x + (k_1 - j_1)\}$$
$$D_6 = \{(x, y)|y \leq x + (k_0 - j_0), y \leq -x + (k_0 + j_0), y \geq x + (k_1 - j_1)\}$$
$$D_7 = \{(x, y)|y \geq x + (k_0 - j_0), y \leq -x + (k_0 + j_0)\}$$
$$D_8 = \{(x, y)|y \geq x + (k_0 - j_0, y \leq -x + (k_1 + j_1), y \geq -x + (k_0 + j_0)\}$$
$$D_9 = \{(x, y)|y \leq x+(k_0-j_0, y \geq -x+(k_0+j_0), y \leq -x+(k_1+j_1), y \geq x+(k_1-j_1)\}.$$

Theorem 3.9. *Let T_0 and T_1 be any two points in $\mathbb{Z}^2$ with coordinates (j_0, k_0) and (j_1, k_1), respectively. Assume that nodal patterns P_i were formed by square waves, i.e. $i = |d_{\Box}(T_0, \rho) - (d_{\Box}(T_0, T_1) + d_{\Box}(T_1, \rho))| \pmod s$. For any point $\rho \in \mathbb{Z}^2$ the membership to one of the following sets $\{D_1, D_3, D_5, D_7\}$, $\{D_2, D_4, D_6, D_8\}$ or $\{D_9\}$ is uniquely identified by a number of distinct nodal patterns P_i in the Moore neighbourhood of ρ.*

Proof. In a system with two transmitters, T_0 and T_1, nodal patterns are formed by broadcasting the square wave from T_0 which, once reached by the wave, will then be broadcast by T_1. The main observation is that broadcasting a square wave generates quadrants defined by the lines $x = y$ and $x = -y$, assuming that the transmitter is the origin, whereby within each quadrant the front of the wave expands such that each element of the orthogonal axis, to the one wholly contained by the quadrant (i.e. x,-x,y,-y), within the quadrant will contain the same member of the alphabet as all the others. This defines the direction of the wave and dictates the structure of the neighbourhoods for the automata here depicted in Figure. 3.4.

It is shown that the 3 differing combinations of directions of the waves result in the same number of differing nodal patterns and that the number of distinct combinations of differing wave directions equate to the number of distinct modes that appear on the plane.

The nodal pattern at some point $\rho = (x, y)$ is given in Definition. 3.8 as

$$i = |d(T_0, \rho) - (d(T_0, T_1) + d(T_1, \rho))| \pmod{s}$$

and the distance function for the square wave is given as

$$d_{\square}((x_1, \dots, x_n), (y_1, \dots, y_n)) = \max_{1 \leq i \leq n} \{|x_i - y_i|\}, (x_1, \dots, x_n), (y_1, \dots, y_n) \in \mathbb{Z}^n \ .$$

The statement of the theorem is proved by showing the form of the neighbours in the four distinct cases and illustrating their differences, they cannot be the same assuming some sufficient alphabet, naturally a single alphabet will render the plane in a singular nodal pattern.

If ρ, $\rho_1 = (x + 1, y)$ and $\rho_2 = (x, y + 1)$ are in the previously defined areas $\{D_2, D_6\}$ then according to the distance function in Definition. 3.8 the difference in distance between $d(T_0, \rho)$, $d(T_0, \rho_1)$ and $d(T_0, \rho_1)$ corresponds to the difference in the abscissa and as such $d(T_0, \rho) + 1 = d(T_0, \rho_1)$ and $d(T_0, \rho) = d(T_0, \rho_2)$. Similarly distances for T_1 are restricted to differences in the ordinate resulting in $d(T_1, \rho) = d(T_1, \rho_1)$ and $d(T_1, \rho) + 1 = d(T_1, \rho_2)$. As such $i_\rho = i_{\rho_1} + 1$ and $i_\rho = i_{\rho_2} - 1$. The full neighbourhood may now be inferred:

$i_\rho - 2$	$i_\rho - 1$	i_ρ
$i_\rho - 1$	i_ρ	$i_\rho + 1$
i_ρ	$i_\rho + 1$	$i_\rho + 2$

Cases $\{D_8, D_4\}$ are seen to be the same via the symmetry of the two groups.

By the same reasoning the area D_9 is explored noting that in such an area the distance function shows the abscissa to be greater than or equal to the ordinate for both $T_{\{1,2\}}$ resulting in $d(T_0, \rho) = d(T_0, \rho_1)$ and $d(T_0, \rho) = d(T_0, \rho_2)$ and for T_1

the following $d(T_1, \rho) = d(T_1, \rho_1)$ and $d(T_1, \rho) = d(T_1, \rho_2)$. This give nodal pattern differences as $i_\rho = i_{\rho_1} + 2$ and $i_\rho = i_{\rho_2}$.

Further more the areas $\{D_3, D_7\}$ which, by the same reasoning, the distance functions from $T_{\{1,2\}}$ are again bounded by the abscissa. Distances for $d(T_0, \rho) + 1 = d(T_0, \rho_1)$ and $d(T_0, \rho) = d(T_0, \rho_2)$. Similarly distances for T_1 are restricted to differences in the ordinate resulting in $d(T_1, \rho) = d(T_1, \rho_1)$ and $d(T_1, \rho) + 1 = d(T_1, \rho_2)$. .This results in nodal pattern neighbourhoods of $i_\rho = i_{\rho_1}$ and $i_\rho = i_{\rho_2}$. By symmetry, neighbourhoods in areas $\{D_1, D_5\}$ are equivalent.

□

3.2.2 Nodal Patterns Resultant from One Source

A pattern produced by both types of wave from a single source gives the possibility to partition the plane in to four distinct partitions. As such it is possible to elect, Proposition. 3.5, vertical, horizontal (set L_0) and diagonal lines (set L_1) starting from a single source by transmitting square and diamond waves after a constant delay.

Proposition 3.10. *Given a vertex, $T_0 = (j, k)$ on $\mathbb{Z}^2$, it is possible to elect a set of points $L_0 = \{(x, y)|y = k, x \in \mathbb{Z}\} \cup \{(x, y)|x = j, y \in \mathbb{Z}\}$ or the set $L_1 = \{(x, y)|y = x + k - j \in \mathbb{Z}\} \cup \{(x, y)|x = -y + k + j \in \mathbb{Z}\}$ in linear time as by Definition. 2.11.*

Proof. Without loss of generality we may assume that the source, T_0, is at $(0, 0)$ and that both the square and diamond waves are broadcast at the same time. Whilst simultaneous broadcast is not possible all values are over modulo equivalent to the size of the alphabet. Hence any delay between the broadcast of the two waves that is equivalent to the size of the alphabet will reduce to 0 resulting in equations that will be the same as if a simultaneous broadcast had taken place. Patterns formed by the transmission of a square wave, shown in Figure. 3.5 a), then a diamond wave, shown in Figure. 3.5 b), can now be expressed as the

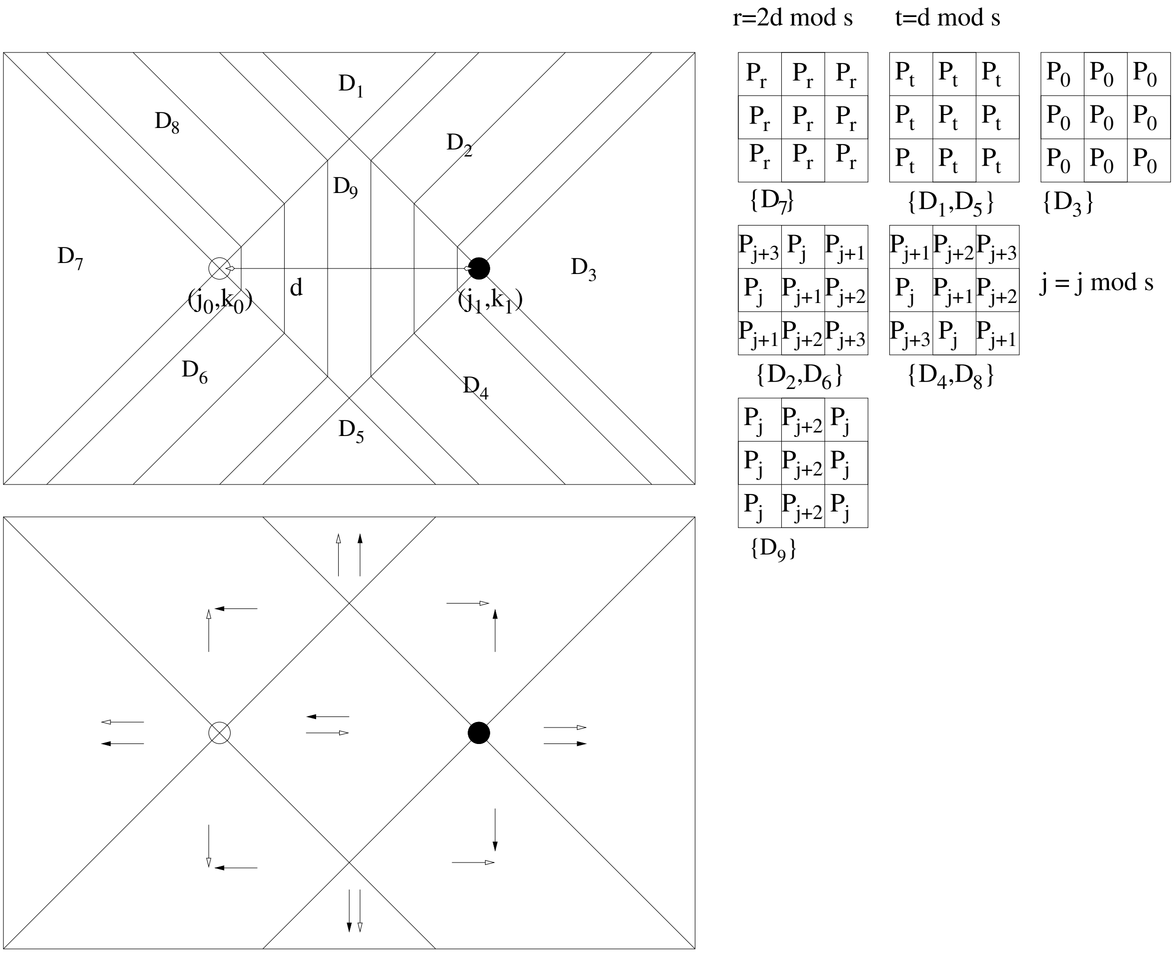


FIGURE 3.4: The distribution of three distinct nodal patterns in case of two 'transmitter', where $s = 4$. The bottom diagram showing the 'motion' of the waves, that is it is in these directions that the indices of the messages increases. The top diagram indicates the segmentation and direction of the resulting nodal patterns where the series of diagrams to the right illustrate the categorisation of patterns that are given in the sector D_i

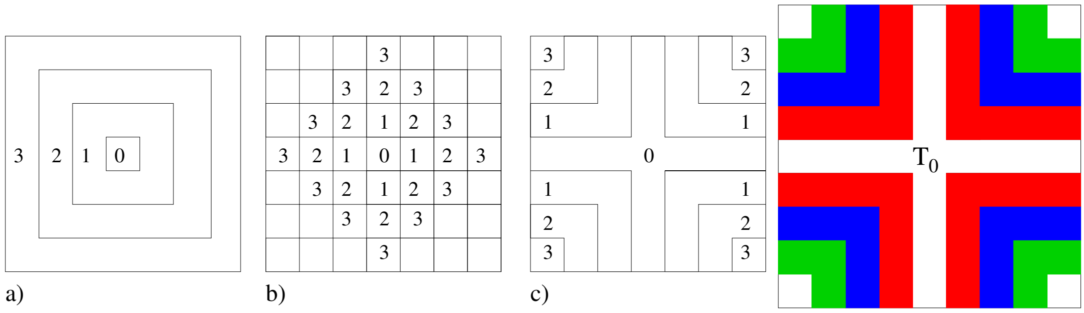


FIGURE 3.5: Figures a) and b) show the nodal labellings of the automata from the square and diamond waves respectively which generate the final nodal pattern shown in c). The far right diagram shows the final cross pattern that can be formed by from a single source by transmitting the square and then diamond wave.

difference in time between its activation by the diamond wave the point at which its nodal pattern is fixed by the square wave or $P_{d_\Diamond((0,0),(\rho_1,\rho_2))-d_\Box((0,0),(\rho_1,\rho_2))}$ for any point $\rho = (\rho_1, \rho_2)$ where $d_\Diamond$ is the distance function for the diamond wave and $d_\Box$ is the distance function for the square wave. If points are of the form $\rho = (0, \rho_2)$ for $\rho_2 \in \mathbb{Z}$ or $\rho = (\rho_1, 0)$ for $\rho_1 \in \mathbb{Z}$ then $P_{d_\Diamond(0,\rho_2)-d_\Box(0,\rho_2)} = P_{\rho_2-\rho_2} = P_0$ or $P_{d_\Diamond(\rho_1,0)-d_\Box(\rho_1,0)} = P_{\rho_1-\rho_1} = P_0$ so all points in L_0 have been elected as P_0, shown in Figure. 3.5 c). This notion can be observed through the composition of the labelled numbers in diagrams Figure. 3.5 a) and Figure. 3.5 b) to get, after addition and through modulo 4 a diagram that resembles Figure. 3.5 c).

The elected nodes will be distinct from the nodes that surround them as can be seen if r is substituted for 0 or $\rho = (r, \rho_2)$ for $\rho_2 \in \mathbb{Z}$ or $\rho = (\rho_1, r)$ for $\rho_1 \in \mathbb{Z}$ then $P_{d_\Diamond(r,\rho_2)-d_\Box(r,\rho_2)} = P_{r+\rho_2-\rho_2} = P_r$ or $P_{d_\Diamond(\rho_1,r)-d_\Box(\rho_1,r)} = P_{r+\rho_1-\rho_1} = P_r$ for $\rho_,\rho_2 > r$ or $P_{d_\Diamond(r,\rho_2)-d_\Box(r,\rho_2)} = P_{r+\rho_2-r} = P_{\rho_2}$ or $P_{d_\Diamond(\rho_1,r)-d_\Box(\rho_1,r)} = P_{r+\rho_1-r} = P_{\rho_1}$ for $\rho_1, \rho_2 < r$. Showing the elected points in L_0 to be distinct where $\rho_1, \rho_2, r \neq 0$.

The election of the set L_1 requires the neighbourhood to be checked through the use of a diamond node wave originating from the initial point T_0. The distinct neighbourhoods of the diagonal elements which are detected with a diamond node wave allows the election of all points in L_1, two diagonal lines meeting at point T_0. □

Theorem. 3.9 and Proposition. 3.10 will feature prominently in the following chapter on algorithms. In fact it is quite safe to say that almost all work on algorithms will be built on the principles that are outlined by this Theorem and proposition. The ability to generate straight lines where no information is known by the automata about the orientation, origin or indeed anything outside of the local reception of messages is essential if the model is to be used to generate partitions, patterns and lines which are to be utilised for geometric computation. It is also of paramount importance that once these lines and partitions have been constructed that it is possible for the automata to be able to elect themselves to the correct partition or part of the line using only their minimal capability to receive transmissions from their surrounding neighbours and to store distinct messages as a set. It can be noted that such minimal abilities are indeed the most minimal class of such networked automata [63].

Whilst, even with the examples that are given here, it is possible to construct complex algorithms, in this case the leader election algorithm that shall be described in detail in the next chapter, once this is generalised to a variable radius transmission then it is possible to construct polygonal partitions with a variety of sloped sides as shall be seen in the categorisation of the discrete circle and its properties in later chapters. Here, even with only the most basic of principles it has been seen, and will be seen, that it generates a useful and usable set of primitives for the construction of algorithms.

Chapter 4

Distributed Algorithms in the Broadcasting Automata

Previous chapters have explored many of the primitives of broadcasting automata, the message passing primitives and many of the forms that are produced as a series of nodal patterns over the plane. In this chapter those primitives shall be assembled in to a variety of algorithms. Some translated from the cellular automata cannon, with respects to the firing squad synchronisation problem, and others taken from the distributed algorithms model, in the from of finding the centre of a discrete disc which can be considered to be a solution to the leader election problem that is commonly found in the distributed computing world [31] and constitutes the central result of the chapter. The rest of the assorted algorithms are general 'helper' algorithms that will be found to be useful in a large majority of the cases when constructing algorithms in this model.

The main result of this chapter is the algorithm for the location of the centre of the digital disc where the initial transmitter is located on the edge of the disc. The algorithm uses all of the techniques discussed in the previous chapter, that utilise nodal patterns, to locate the centre of the disc through distributed, geometric computations.

4.1 Basic Algorithms

The following algorithms are considered basic in that they are consider to be the basic building blocks of algorithms, used as helper algorithms, which would most likely be required in the construction of more complex algorithms. In general these algorithms are considered to alleviate some of the restrictions of the model illustrating that even though the model may be technically primitive and computationally simple it is still possible to, through a variety of algorithmic techniques, remove the limitations inherent in the model.

4.1.1 Firing Squad Synchronisation Problem for BA

The Firing Squad Synchronisation Problem (FSSP) originated with J. Myhill in 1957 and is informally described by the following analogy:

> Given a firing squad and knowing that any order given by a general located at one of the two ends needs n units of time to reach the nth soldier, how can we make all the soldiers fire at the same time [70].

The solution to the problem is the construction of a finite state automata that is homogeneous for every element in the one dimensional line and such that the automata is not changed if the size of n is increased. There have been many attempts to minimise two factors of the complexities of the automata, the number of states and the time taken to synchronise the automata.

It has been shown that the lower bound for a solution to FSSP is [71] $2n-2$ for n automata. There have been several advancements in attempting to find a minimal state solution that can solve the problem and a minimal state configuration that can solve the problem in minimal time. It has also been shown that there is no optimum time solution in four states [72] where a five state solution is known to be an open problem [70] and the minimal time solution with the smallest number of states that is known is a six state solution [71].

In this setting the problem is the standard FSSP problem whereby the automata in question are only able to send and receive messages of a single character, as opposed to the usual observation of the neighbours states as presented in the version of the problem which has been tackled by cellular automata. It is possible to solve the FSSP problem in this fashion with an alphabet of four where all automata are modelled the same except for the two end generals which need only a slight modification of their rules.

For the algorithm only two cases need be considered and these are the cases in which the elected points are of an even distance and those which are of an odd distance owing to each recursive sub-problem falling in to one of these categories. The state diagram for the automata is shown in figure. 4.1. This is a translation of Minsky's solution [73] in $3n$ time to the firing squad problem which completes in a time $3n$.

4.1.1.1 Odd Case

The intuition behind this algorithm is that two messages are passed on at 1/3 of the speed of the other and when these meet the centre of the line of automata will have been elected. The first pulse that is sent will travel at a speed of 1 with each of the automata receiving the first a signal immediately transmitting it to the next. The other pulse initiates a series of states implementing a time delay of 3 before sending the second a which propagates this delay of 3 for each automata, after sending this last a the automata resets to the quiescent state, this allows deconstruction in to simple identical sub-problems all starting from the quiescent state. When the first signal, propagating at a speed of one, reaches the general that is opposing the first it sends back a signal, c. When a b signal is propagated to an automata in the delay 3 loop then it becomes a general.

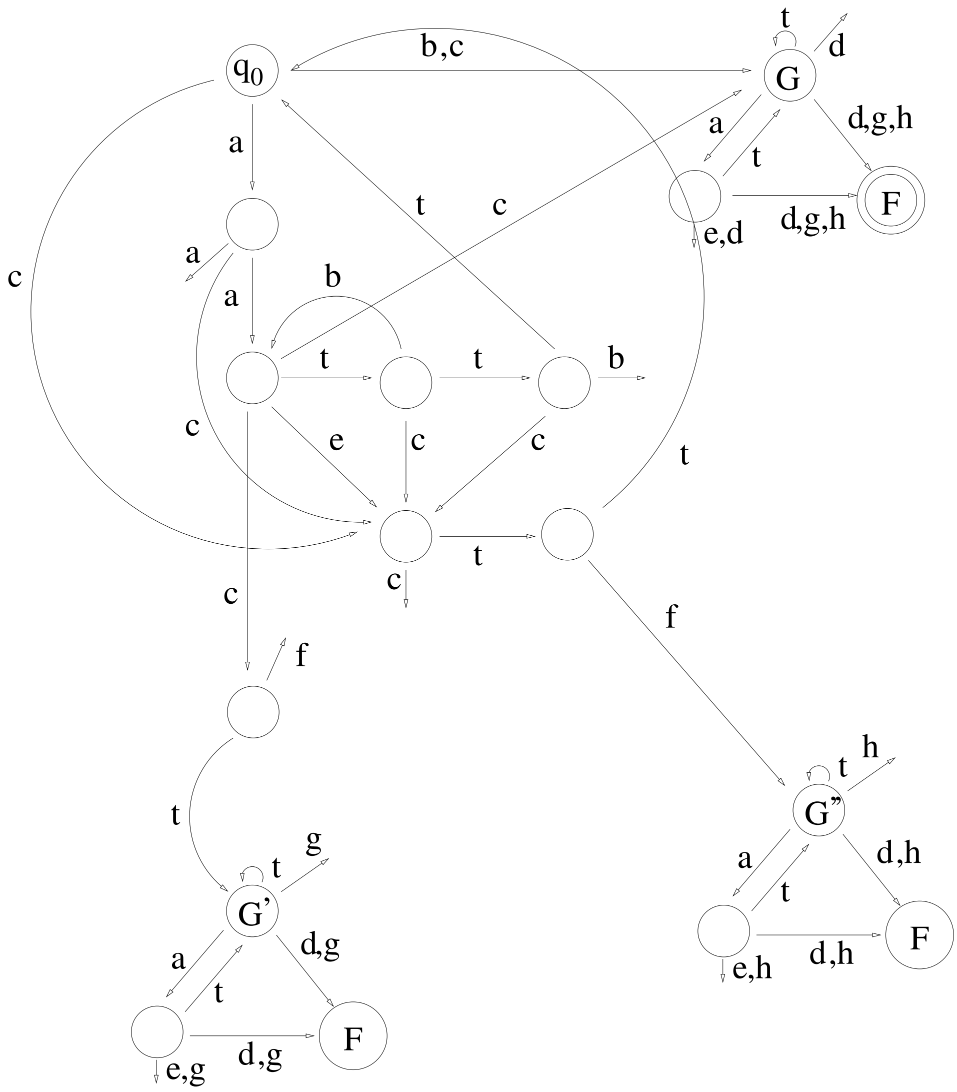


FIGURE 4.1: The above figure represents the state diagram for each of the broadcasting automata involved in the Firing Squad Synchronisation Problem.

4.1.1.2 Even Case

The basic concept here is the same except that this time the automaton in the delayed signal loop will be met with a *c* signal at which point the automaton makes the transition to the state outputting the *f* symbol and after a time delay both automata have been elected as the central automata, the new generals, the leftmost general, G', will be outputting the symbol, g whereas the rightmost will be outputting the symbol, h, from general G''. This is done so that they do not move in to the firing state prematurely by instantly sending each other a d symbol.

4.1.1.3 Firing State

The final firing state is achieved when the generals receive a transmission which makes the transition to the final state. In this case it must be from a general that is emitting either a d symbol or the same symbol as the general symbol it's emitting.

4.1.1.4 Correctness of the Algorithm

The correctness of the algorithm can be derived from the paper by Minsky [73] from which the solution is a translation of. Intuitively the automata constantly divide the set of automata in half creating two new subproblems. Such subproblems are solved recursively, in the same way, until there are two generals next to each other that were not formed in the same even case dividing operation. Here all generals transition to the firing state.

4.1.2 Election of the Edges of BA in $\mathbb{Z}^n$

This algorithm is of the class of algorithms that are considered building blocks for much larger algorithms allowing a circumvention of some the models restrictions in an algorithmic way. Here, the problem of being able to locate the end of a line of automata is tackled. The problem arises because of the way the automata detect their neighbours which is only by being informed of their states. The automata in this way are otherwise directly blind to their neighbours being unable to 'probe' for their neighbours states or even to find out if they exist. The following Lemma may be used in this way to remove this limitation.

Lemma 4.1. *Given a network of finite automata,* (G, Λ, V_0) *with a graph which form a single line and is of size* n *and a single source* v*, such that* $v \in V_0$*, it is possible to elect the automata with no successor in* $(n-k)+3$ *time steps, where* k *is the number of automata from the initiating automaton,* v*, to its closest side, and with an alphabet,* Σ*, of size* $|\Sigma| = 3$*, in the asynchronous model.*

Proof. The proof is given as a constructive argument such that the following algorithm describes a method for any automaton to establish itself as an automaton that is, or is not, on the edge and correctness of the algorithm is given. The algorithm, presented below, assumes that there is some $v \in V_0$ that begins the initial transmission and that the standard form of transmission for broadcast automata is utilised. After use of the following procedure, which is utilised by all broadcast automata, then any such automata which have no successor should have elected themselves as such.

The following algorithm is followed by every automata and provides a method for locating the edge automata in any line of size n with an alphabet of size 3 and is visualised in Figure. 4.2.

1. Automaton v_i receives symbol $s_i \in \Sigma$ activating it at time t
2. For the following three time steps, $t + \{1, 2, 3\}$, it emits symbol $s_{i+1 \bmod 3}$
3. If at time $t + 2$ automaton v_i receives two signals of differing types, s_i and $s_{i+2 \bmod 3}$ then it is not on the edge. Should the v_i at time $t = 3$ receive only signal s_i then the automaton has no successor.

The above algorithm elucidates the observation that those automata on the edge of the graph have no successor and as such it is not possible for them to receive two signals of differing types. Let us assume for contradiction that there is such an element v_i that has no successor but which will not be elected by the above algorithm. It must be that v_i, to have successfully been elected by the algorithm, has received two differing symbols at the same time step. As each automaton is only able to transmit a single symbol during one time step this means that the automaton v_i must have at least two automata in its neighbourhood. As the automaton lies on a line it must be that this automata has both a predecessor and a successor however this is now in contradiction to the premise that this automaton has no successor and so the algorithm must hold. □

Whilst the algorithms may be presented in a single, specific model, asynchronous, it is possible to transfer that algorithm to the opposing model, synchronous, by the result presented in Proposition. 2.12.

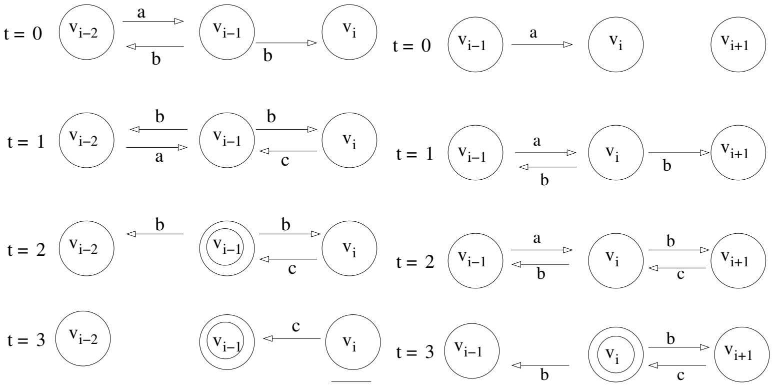


FIGURE 4.2: The detection of a none edge case, that has two neighbours (left), and an edge case, with only one neighbour (right)

4.1.3 Election of the Shortest Branch of a Tree

The following algorithm presented in the Lemma shall be important in the case of electing the centre of a digital disc but as the notion of distance is intrinsically linked to broadcasting automata through the notion of nodal patterns it may very well be considered another example of a fundamental algorithm.

Lemma 4.2. *It is possible to elect the automaton with no successor of the shortest branch of a tree with a single node at the root with k branches, such that no other node in the tree branches other than the root node, in time $3R$, where R is the length of the shortest branch(es) with a distinct alphabet of $|\Sigma| = 2$.*

Proof. The following algorithm provides a constructive proof of the existence of such an algorithm. It is assumed that there is an initial automaton $v_0 \in V_0$ that is at the root vertex of the graph.

1. From the root vertex v_0 the symbol a is broadcast to all neighbours of v_o, propagating through the system as defined by the message passing algorithm for broadcast automata.

2. Once the wave reaches the final vertex in each branch it is broadcast back up towards the root vertex after 2 time steps waiting for the refractory period of the previous vertex to expire.

3. After the root receives the first of the a symbols it begins to broadcast symbol b which is only accepted by those automata which have received two a symbols, which are the automata of the shortest branch(es).

4. Upon receiving symbol b those automata that have already received a, a will be elected as the shortest branch(es).

   All other automata which have at that moment only received a single a will have rejected the b symbol and so are not elected.

The correctness of the algorithm is given by the lengths of the paths in which the transmissions must traverse. Given a set, S, of k branches such that each branch is of length $p_0 \leq p_1 \leq ... \leq p_{k-1}$ there must be some set of least paths which will be denoted S_{min} and contain only those branches that are of equal length to p_0 the branch of least length. Given that the messages are passed at constant speed, and as such are synchronised throughout the system, it must be that each path is traversed in time $2 \cdot p_i$ and those that reach v_0 first, after the initial transmission from v_0, must have taken the shortest path, here given as S_{min}. As only those automata which have received the symbol a twice are elected by the b symbol, other automata, in the set $S \setminus S_{min}$, will reject the symbol, and so only those in the set S_{min} will be elected by b. □

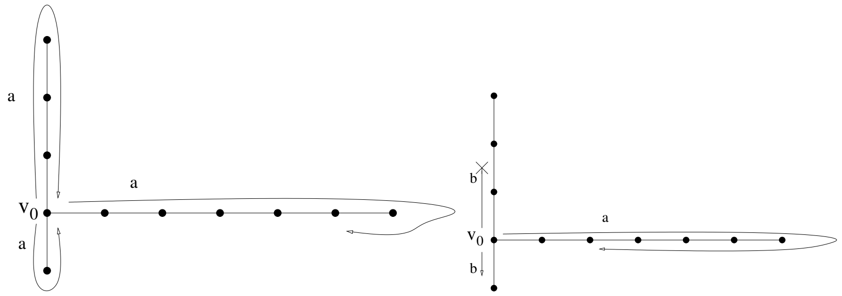


FIGURE 4.3: The above two figures show the algorithm in action.

4.2 Geometric Computations with BA

The following section showcases the power of broadcasting automata that complex algorithms can indeed be derived from such a simple model with low computational abilities and a simple message passing schema. The model is used to solve a problem that could be stated as one of the most fundamental problems in distributed algorithms [31] and also one of the most often tackled problems in swarm robotics [7].

4.2.1 Finding the Centre of a Discrete Disc

A set ζ of points in $\mathbb{Z}^2$ is a **digital disk** if there exists a Euclidean circle, with a centre at an integral point, that encloses the pixels of ζ but excludes its complement. Let us consider a model of broadcasting automata on a digital disk which has a diameter D. We define a procedure for finding the centre of the digital disk in linear time using the notion of waves as described in previous sections. The problem is now posed as such;

Problem 4.3. Beginning from any point on the circumference of the circle it is possible to find the centre of the digital disk as a single point or as a set of two points, depending on whether the radius of the digital disk is odd or even, respectively.

In this section we abbreviate the partitions previously mentioned to $a = \{D_9\}$, $b = \{D_1, D_3, D_5, D_7\}$ and $c = \{D_2, D_4, D_6, D_8\}$. The algorithm for finding the centre can begin from any arbitrary point, T_1, on the edge of the digital disc and is implementable in both asynchronous and synchronous models.

Depending on the location of the initial point, one of three algorithms is applicable. Finding the correct algorithm to apply is reduced to checking the initial point's neighbourhood to one of three possible sets in the following way.

Definition 4.4. Eight points $\{0, 1, ..., 7\}$ on the circumference of a digital circle, ζ corresponds to the following eight angles $\{0, 45, 90, 135, 180, 225, 270, 315\}$.

Lemma 4.5. *Given an automaton on the edge of ζ it is possible to check the automaton's membership to one of three sets:* $\{0, 2, 4, 6\}$*,* $\{1, 3, 5, 7\}$ *and all other points on the edge of the circle in a time* $O(D)$ *for both models.*

Proof. From automaton T_1, on the edge of ζ, Proposition. 3.10 is applied via the transmission of a square then diamond wave, resulting in horizontal, vertical lines of elected automata and electing at most two paths. All elected automata on the edge of the disc, ζ, become new points T_N which is a set of points encompassing up to three automata, $\{T_2, T_3, T_4\} \in T_N$. As soon as the automaton or automata on the edge of the disc, ζ, have been elected by T_1, the automata, now denoted as T_N, begin transmission of a square wave. As the transmission of these waves from all automata in T_N, may occur simultaneously on the disc, points at which waves meet each other proceed no further on the disc, due to the automata's inability to receive and broadcast at the same time, cancelling each other. Points of wave cancellation are shown as dotted lines in Figure. 4.4 and Figure. 4.5. The partitions formed by the transmissions from new points in T_N can now be detected by the initial point T_1 through the transmission of a neighbourhood detection wave which gives nodal patterns to automata through the transmission of its own square wave and causing neighbouring automata to transmit their states which allows the detection of T_1's neighbouring nodal patterns. By Theorem. 3.9, possible neighbourhood partitions for the initial point T_1 are now be categorised as $\{a\}$,$\{a, c\}$ and $\{a, b, c\}$ which are the points $\{0, 2, 4, 6\}$ and $\{1, 3, 5, 7\}$ and all other points respectively. The procedure requires only three waves of transmissions, each wave requires an amount of time that is no more then the diameter of the circle as well as some constant time between transmissions and the constant time for the neighbourhood recognition.

□

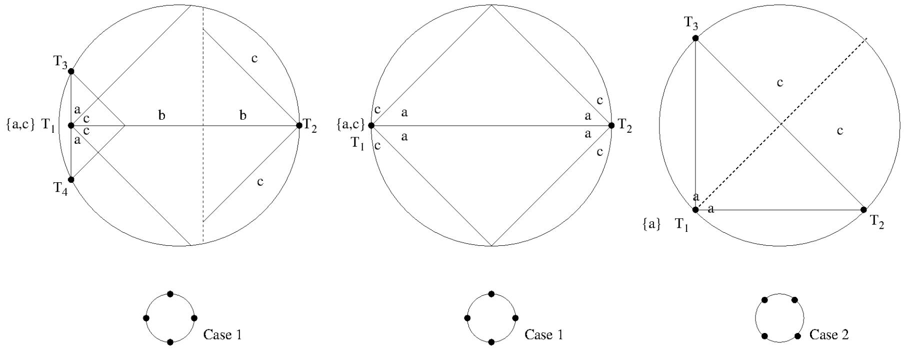


FIGURE 4.4: The above Figure shows two of the three possible cases stemming from the five possible variants that require differing solutions based on their location. The two differing sets of the 8 points and those points which lay in none of these. The two diagrams for Case 1 correspond to the situations whereby, for case 1, T_1 generates a 3-branched tree with two equidistant branches or a single chord respectively.

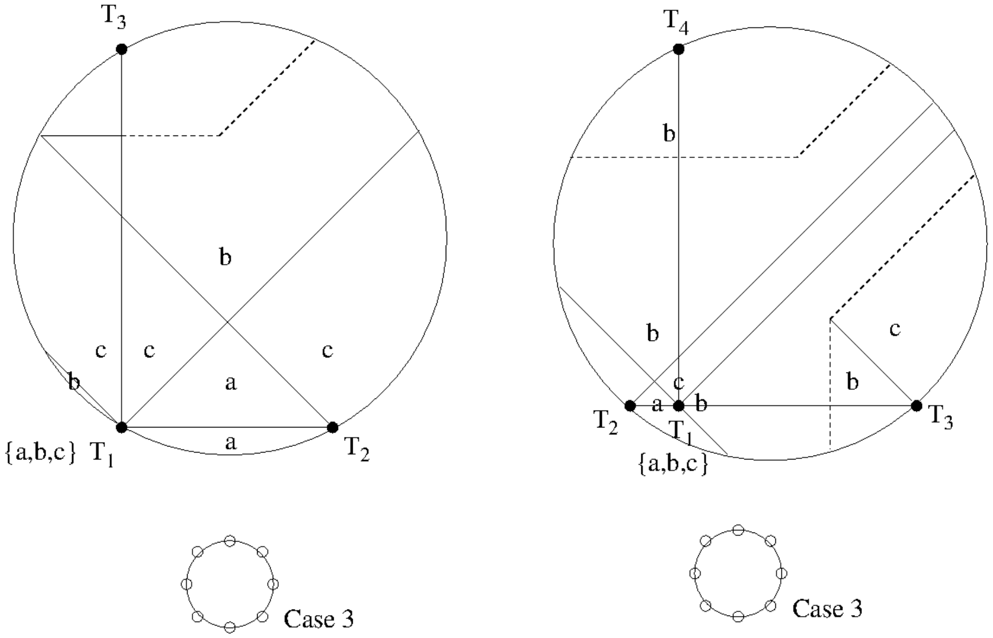


FIGURE 4.5: The above figure shows the two possible variations of the final case of the three cases. For case 3 the situations are those such that T_1 generates a 2- or 3-branched tree respectively.

4.2.2 An Algorithm for Locating the centre of a Digital Disk

There are many edge cases to consider in the construction of the algorithm that is presented in this section. This part of the algorithm may be considered almost a meta algorithm or a ‘switch’ which uses Lemma. 4.5 in order to find which of the algorithms is applicable due to variations in the formations that may be found on the edges of the circle, where the initial transmitter is found.

1. An automaton on the edge of the disc ζ, T_1, checks its location by the creation of the unique local neighbourhood sets: $\{a\}$, $\{a, c\}$ or $\{a, b, c\}$, see Lemma. 4.5.

2. In case of neighbourhood set $\{a, c\}$ apply the algorithm for case 1. In case of neighbourhood set $\{a\}$ apply algorithm case 2.

3. In the case of neighbourhood set $\{a, b, c\}$, T_1 is the root of a tree with two or three branches. The third branch may appear if the automata, T_1, finds itself on a ’ledge’, such that there are automata on three sides of its Von Neumann neighbourhood, formed from the digitisation of the circle. The end point of the shortest branch of such tree, placing the automata in a position whereby there are only two automata in its Von Neumann neighbourhood, is found by Lemma. 4.2 which is relabelled T_1 and then apply case 3.

Cases 1 and 2 are basic because the location of the point T_1 is known exactly. The least trivial case is Case 3 where further partitioning is required for locating the centre. **Algorithm for Case** 1 **(set** $\{0, 2, 4, 6\}$**)**

1. T_1 sends message m_0 to T_2 through the chord which will be sent back from T_2 after some constant delay $k = |\Sigma|$.

2. T_1 sends message m_1 which has a delay of 3 after a constant delay $k = |\Sigma|$.

3. The automata on the chord elected through receipt of both message m_0 and m_1 at the same time will be the centre of the digital disc ζ.

Algorithm for Case 2 (set $\{1, 3, 5, 7\}$)

1. A new point T_4 is generated along the diagonal through the use of diamond neighbourhood detection wave as described in Proposition. 3.10.
2. T_1 sends message m_0 to T_2 through the chord which will be sent back from T_2 after some constant delay $k = |\Sigma|$.
3. T_1 sends message m_1 which has a delay of 3 after a constant delay $k = |\Sigma|$.
4. The automata on the chord elected, through receipt of both message m_0 and m_1 at the same time, will be the centre of the digital disc ζ.

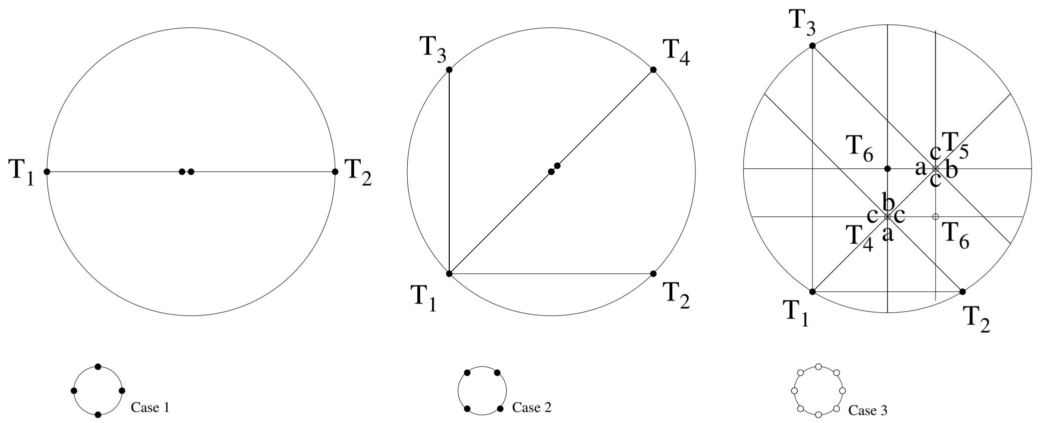


FIGURE 4.6: Constructions required to find the centre of the circle for the three differing cases. Centres are indicated by black dots.

Algorithm for Case 3

1. Point T_2 is identified as the shortest chord of the tree constructed from T_1 via Lemma. 4.2.
2. Point T_3 is identified as the longest chord of the tree constructed from T_1 by broadcasting a signal c only to those automata that have received two a's but no b's after Lemma. 4.2, the longest chord.

3. Transmissions from T_3 followed by T_1 elect the point T_5, the only point on the digital disc that has a neighbourhood containing the nodal patterns $\{a, b, c\}$.

4. Transmissions from T_2 followed by T_1 elect the point T_4, the only point on the digital disc that has a neighbourhood containing the nodal patterns $\{a, b, c\}$.

5. Application of proposition. 3.10 from T_4 and T_5 will generate horizontal and vertical lines. The points at which these two line cross are elected as T_6. Only one of the two points marked T_6 will be in partition a given by the initial construction of T_1 and T_3, this is the centre point of the circle.

Theorem 4.6. *It is possible to find the centre of a digital disc ζ with diameter D starting from a point on the circumference of ζ, T_0, in both models, in $O(D)$ time, where the notion of complexity is given in Definition. 2.11.As there are a finite number of passing of waves which all time-bounded by at most $O(D)$, the algorithm cannot exceed linear growth by D.*

Theorem. 4.7 directly follows from the construction of the next algorithm for electing the elements of the inscribed square and also holds when starting from a point is on the circumference of ζ.

An Algorithm for electing elements of the inscribed square

Theorem 4.7. *Given an initial transmitting node in the centre of a digital disc. It is possible to elect automata forming an inscribed square in time $O(D)$, where D is the diameter of the digital disk.*

Proof. 1. From point P_1 transmit two waves (square and diamond) to elect the four points A_1, A_2, A_3 and A_4. This follows from Theorem. 3.10 whereby the already mentioned subset is reduced by intersection with the set of those points on the edge of the circle. (see Figure. 4.7).

2. A square wave is transmitted from the points A_1, A_2, A_3 and A_4 as elected in the previous step of the algorithm. The waves form four squares, shown as S_1, S_2, S_3, S_4 but do not intersect each other as once the waves meet all transmission halts. The points at which the waves meet are exactly those that are along the diagonal lines that intersect the central point P_1.

3. A wave is now sent from P_1 forming nodal patterns throughout the circle. It can be noted now that the patterns that form, for each pair, (P_1, A_i), are those which are given in Theorem. 3.9.

4. Another wave is now propagated from P_1 which informs the automata of their neighbours and allows the automata to place themselves in the set D_9 if it is within the inscribed square.

Note that the complement of the square will also have a distinct pattern of type $\{D_2, D_4, D_6, D_8\}$ again given by Theorem. 3.9.

It is noted that the bounds on complexity are again calculated based on the notion of complexity as given in Definition. 2.11 such that linearity, n, is considered to be for some coefficient of messages passed to every member of the Broadcasting Automata. The complexity is derived from the liner time it takes to pass the wave, a wave pass is more formally a transition from one configuration to another, and the maximum time that this transmission can take is equivalent to the diameter of the disc. Any multiple of D still results in an $O(D)$ complexity as previously noted. □

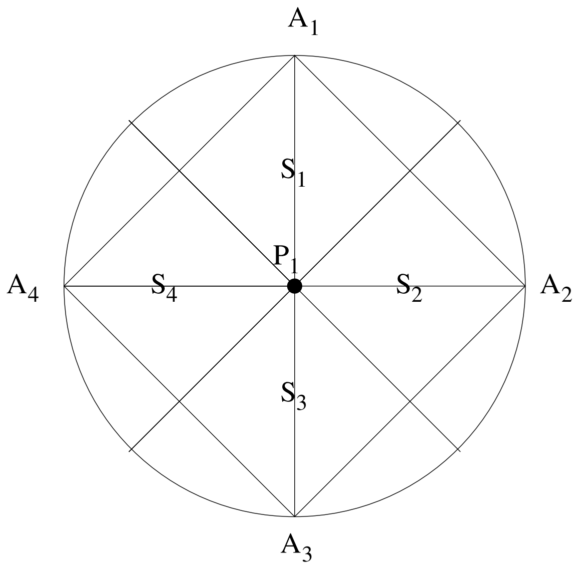


Figure 4.7: Figure corresponds to production of the inscribed square from centre point.

Chapter 5

Transmission Radii and their Wave Formation in $\mathbb{Z}^2$

There are two main variables in broadcasting automata, the words that are transmitted through the lattice of automata forming nodal patterns in the plane as they intersect and the size of the transmission radius, altering the number of neighbours that receive that signal in a single time step. In this chapter a characterisation of the shape of the discrete circle will be given. From this information it, in conjunction with the work on the resultant nodal patterns formed by functional combination, is possible to derive a full characterisation of the nodal patterns as they fall on the plane under broadcasting automata.

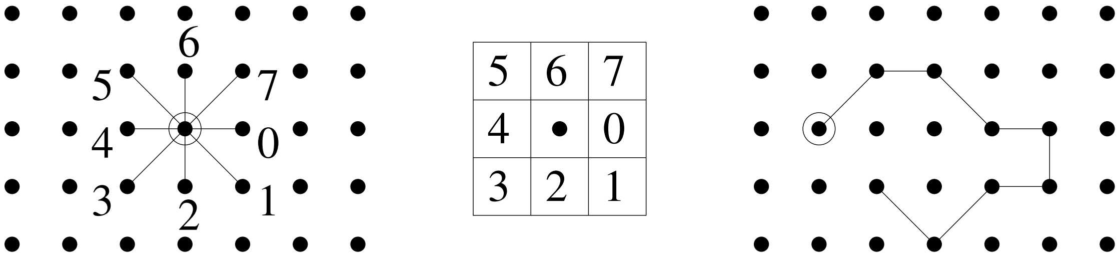
5.1 Geometrical Properties of Discrete Circles

This section discusses the geometrical properties of discrete circles on the square lattice and its chain coding and their manipulation. Chain code representation of circles are characterised and results given in [74] are expanded. Characterisation of the circle is given with respects to chain coding and the ordering of line segments which make up faces of the resulting discrete polygons are given.

5.1.1 Chain coding

The method of chain coding was first described in [52] as a way in which to encode arbitrary geometric configurations where they were initially used, as they shall be here, to facilitate their analysis and manipulation through computational means. At the time there were many questions about the encoding of shapes and the methodologies which should be used to encode such shapes and chain coding presented an schema which was simple, highly standardised and universally applicable to all continuous curves. Whilst the definition given here differs from the definition given in [52] it only differs for convenience when discussing discrete discs in the first octant allowing the use of only the numeric symbols 0 and 1 instead of 0 and 7 as Freeman suggested. This gives the following definition of chain coding that shall be used throughout.

Definition 5.1. From a starting point on the square lattice a **chain code** is a word from the alphabet $\{0, 1, 2, 3, 4, 5, 6, 7\}$.



5	6	7
4	•	0
3	2	1

FIGURE 5.1: (Left) Showing the eight possibilities of motion from the single point of origin, circled, as they may be traced out on the plane (Centre) Shows the eight possible points of motion from a central point and (Right) gives and example of an arbitrary line traced out on the lattice with a chain code of form 70102435 originating from the circled point.

From some starting point chain codes may be used to reconstruct a shape, S, by translating the code's motion on to the lattice where 0 indicates positive movement along the $x - axis$ with increasing values moving clockwise through all 8 possible degrees of motion on the lattice, labelled from 0 to 7 respectively. An integral shape, S, can be encoded using an inverse method whereby the shape is traced from some starting point and the motion from one point to the next connected point in the shape is recorded as a chain code. For example, a straight line would be represented by the infinite repetition of a single digit such as 00000000000000..., a line that satisfies the equation, $x - y = 0$, may be encoded 77777777... or as the example given in Figure. 5.1 and arbitrary line may be encoded numerically as 7010235.

5.1.2 Discrete Circles and their Chain Code representation

The discrete disc is the basic descriptor of a transmission in broadcasting automata. It represents the total set of all automata that could be reached on the square lattice after a single broadcast of radius, r^2. In the following sections reasoning will follow only from the first octant, which can be seen in Figure. 5.2, of the discrete circle which shall now be defined and gives all the information that is required to categorise the discrete disc, its perimeter, but also has a convenient method of chain code construction. The need only to observe the first octant is derived from the symmetry that occurs such that it is possible to take a solution of the first octant and by successive reflection operations about the eight octants generate a full circle [75] under the assumption, which is also made here, that the circles are centred on a integral point on the plane.

Definition 5.2. A discrete circle is composed of points in the $\mathbb{Z}^2$ set $\zeta = \{(x, y)|x^2 + y^2 \leq r^2,\ x, y \in \mathbb{Z},\ r \in \mathbb{R}\}$ that have in their Moore neighbourhood any point in the complement of ζ.

In Definition. 5.2 it is shown that $r \in \mathbb{R}$ however many of these do not produce distinct discrete circles. The set of r that includes all distinct circles are those which form Pythagorean triples which consist of two positive integers $a, b \in \mathbb{N}$, such that $a^2 + b^2 = r^2$.

A simple way to draw the discrete circle is to divide it into octants as can be seen in Figure. 5.3. First calculate only one octant of the circle and as the rest of the circle is "mirrored" from the first octant it can now be successively mirrored to allow the construction of the other sections of the circle. Let us divide the circle into eight octants labelled 1-8 clockwise from the y-axis. The notation $Octant(i)$

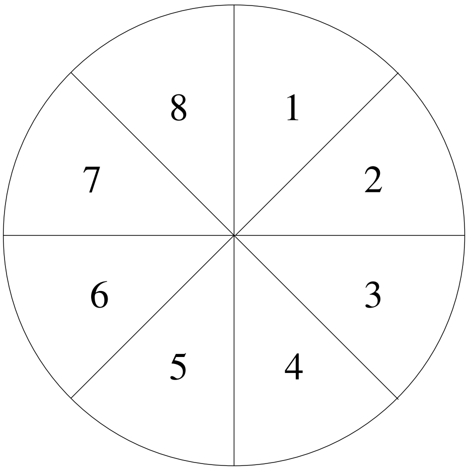


Figure 5.2: An illustration of a circle that has been divided in to eight octants with the labelling of those octants applied accordingly.

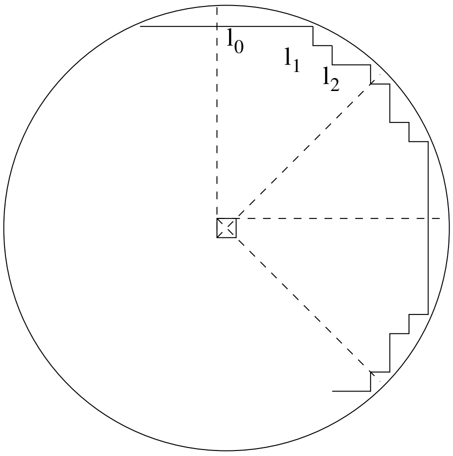


Figure 5.3: An illustration of a discrete disc and its accompanying discrete circle of radius squared 116, where the chain code for the first octant is labelled as $l_0 l_1 l_2$.

is used to refer to octant sector i. If an algorithm can generate the discrete circle segment on one octant, it can be used to generate other octants by using some simple transformations, such as rotation by 90° and reflection around the x-axis, y-axis, and the $+45°$ diagonal lines.

Some basic results are now given about chain codes of such discrete circles. It is possible to show that in the first octant and given the definition of chain coding presented here only two digits, 0 and 1 are required to fully represent the discrete circle.

Lemma 5.3. *A Chain coding w for the circle in the first octant is a word in* $\{0,1\}^*$.

Proof. First let us show that in the first octant for any given point $p_0 = (x_0, y_0)$ on the circle at the point $p_1 = (x_0 + 1, y_0 - 1)$ must also be in the circle as $|p_1| < |p_0|$. When p_0 is in the circle and in the first octant segment such that $x < y$ it follows that the magnitude of p_1 is less than p_0 as $|p_0| - |p_1| = (x_0^2 + y_0^2) - (x_1^2 + y_1^2) = (x_0^2 - x_1^2) + (y_0^2 - y_1^2) = (x_0^2 - (x_0 + 1)^2) + (y_0^2 - (y_0 - 1)^2) = (x_0^2 - x_0^2 - 2x_0 - 1) + (y_0^2 - y_0^2 + 2y_0 - 1) = 2y_0 - 2x_0 - 2 \geq 0$. Since $p_1 = (x_0 + 1, y_0 - 1)$ should be in the circle if $p_0 = (x_0, y_0)$ is on the circle we have that any chain code for the circle in the first octant does not contain a subword "22" as it will lead to contradiction. Moreover in any subword a single symbol "2" should be followed by "0" and can be substituted by 1. □

As only the first octant segment is considered the solutions can be represented using strings consisting of only 0 and 1 where 0 is positive motion along the $x - axis$ and 1 is positive motion along the $x - axis$ and negative motion along the $y - axis$.

Indeed in the generation of such chain codes the following method shall be used which is a modification of the algorithm that is given in [74]. Starting with the equation for the circle $x^2 + y^2 = r^2$ where it is known, by definition, that $r^2 \in \mathbb{Z}$. Now as solutions of the first octant are required and under the assumption that, without loss of generality, the centre of the circle is at the origin it is now possible to replace y^2 with $(r' - k)^2$, where $r' = \lfloor r \rfloor$, such that when $k = 0$ and rearranging for x^2 such that $x^2 = r^2 - r'^2 - k^2 + 2r'k$ the solution to the circle equation gives $x^2 = r^2 - r'^2$ where $r' \leq r$, as such the length of the first solution of such an equation is $\sqrt{r^2 - r'^2}$. Successive values of k give solutions to the length in the x-axis of that particular maximal Pythagorean triangle. All such Pythagorean triangles will give the final solution to the circle.

Here it is necessary to start with some terminology to say more precisely which parts of the chain code are considered to represent specific components in the resulting polygons. It will become much clearer later on why such distinctions are required as a single side of the polygon may differ in its chain code categorisation. The first of these categories of chain code, the most basic, is given here.

Definition 5.4. A **chain code segment** is a subword of a chain code which is either a word $0^{|s_0|}$ or $10^{|s_i|-1}$.

A discrete circle, u, is composed of chain code segments where each segment is represented as u_i, i.e $u = u_0u_1u_2...u_n$. Chain code segments can be expressed using a power notation for the number of repetitions e.g 100100 may be rewritten as $(100)^2$ and if $u = u_0u_1u_2u_3 = (00)(100)(100)(10) = (00)^1(100)^2(10)^1$.

It is possible to map the chain code, u, in $Octant(1)$, to the others octants through a function that maps the alphabet of the chain code $\{0, 1\}$, to code to a new code in another octant segment. A chain code of a circle in an $Octant(n)$ can be constructing from an $Octant(1)$ first by applying the following mapping $\{0, 1\} \to \{n-1 \mod 8, n \mod 8\}$, where $0 \leq n \leq 7$. Then for the $Octant(n)$, where $n \equiv 1 \mod 2$, or n is odd, the code and the mapping itself must be reversed such that $\{0, 1\} \to \{n \mod 8, n-1 \mod 8\}$. For the rest of this paper we will only consider chain codes in the first octant.

Proposition 5.5. *A circle in the first octant can be chain coded as follows:*

$$0^{|s_0|}10^{|s_1|-1}10^{|s_2|-1}\ldots10^{|s_i|-1}\ldots10^{|s_{r'^2/2}|-1}$$

where,

$$s_i = \{x|r^2 - r'^2 + 2(i-1)r' - (i-1)^2 < x^2 \leq r^2 - r'^2 + 2ir' - i^2, x \in \mathbb{Z}\}$$

r is the radius and r' is the floor of the radius.

Proof. It follows from [74] where such a method of chain coding is introduced. □

Proposition 5.6. *Chain code segments, following the schema*

$$0^{|s_0|}10^{|s_1|-1}10^{|s_2|-1}\ldots10^{|s_i|-1}\ldots10^{|s_{r'^2/2}|-1}$$

have the following constraints to their lengths in the discrete circle:

$$\lfloor\frac{s_i - 1}{2}\rfloor - 1 \leq s_{i-1} \leq s_i + 1\ .$$

Proof. It, again, follows from [74] where this property is proved. □

5.1.3 Line Segments

As previously noted terminology must be introduced to distinguish the varying parts of the chain code and in order to talk about what they represent. In order to characterise the discrete circle two notions are introduced, *Line Segments* and *Gradients of Line segments.*The following definition for the gradient gives a method for extracting such information from the series of digits that construe the line as a word.

Definition 5.7. A **gradient**, $G(u)$, of a chain code subword in $\{0, 1\}$, u, is given by $G(u) = \frac{\#_1(u)}{|u|}$, where the function $\#_1$ returns the number of $1's$ found in the chain code and $|u|$ is the length of word u.

Line segments are in reference to actual lines that these sections of chain code would represent on the plane. That is if one were to draw a line and then attempt to translate this line in to a chain, under the assumption that the gradient of the line is rational, for reasons which can be seen in the definition of a gradient, it would be done via a periodic series of chain code segments. Taking a single period of this chain all the information that is required to identify and reconstitute this line to any length is now known. In this way the name may interpreted quite literally as a segment of a line in chain coded form such that any repetition of this object produces a line. With this in mind it will naturally be of benefit to be able to know a little more about the line that has been chain coded as a line segment.

Definition 5.8. A **Line segment**, $l_i = u_j u_{j+1} ... u_{j+n}$, is the shortest contiguous subsequence of chain code segments which maintain an increasing gradient such that $G(l_i) < G(l_{i+1})$, $\forall i > 0$, where the last chain code segment in l_i, $u_j + n$, is contiguous to the first chain code segment in the next line segment, that is, $l_{i+1} = u_{j+n+1} u_{j+n+2} ... u_{j+n+m}$.

As, ultimately, all of the discrete discs that are discussed here, and, more importantly, that are possible to construct, are polygons, such polygons are composed of

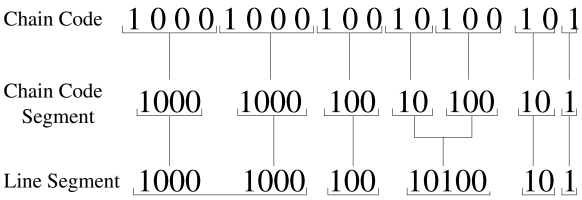


FIGURE 5.4: Showing the the relationship between chain codes, chain code segments and line segments.

line segments which represent the sides of the convex shape on the plane. Further, when categorising the repeated broadcast of such discrete discs, later discussed as composition, the shapes that are generated are known to be similar, in the strict geometric sense. A definition for chain codes, translating from the geometry of the Reals, is given.

Definition 5.9. For any two chain codes, u, v, comprising line segments such that, $u = l_0 l_1 ... l_m$ and $v = l'_0 l'_1 ... l'_n$, for $m = n$, given by Definition. 5.8, are similar (geometrically) if there is a constant $k \in \mathbb{N}$ such that $u = l'^k_0 l'^k_1 ... l'^k_n$.

In this way it is possible to say that two shapes are similar in a geometric sense such that they contain the same number of distinct line segments but the number of each of these line segments is different by some constant.

With these definitions stated it is possible to begin the categorisation of the set of discrete circles with the first observation which extends an already known notion about the chain code of the discrete disc. Following [74] it is known that in the chain code u any chain code segments with increasing lengths, such that $|u_i| < |u_{i+1}| < ... < |u_{i+n}|$, may increase it by at most 1 from the length of a previous segment, i.e. $|u_{i+1}| \leq |u_i| + 1$ for all i.

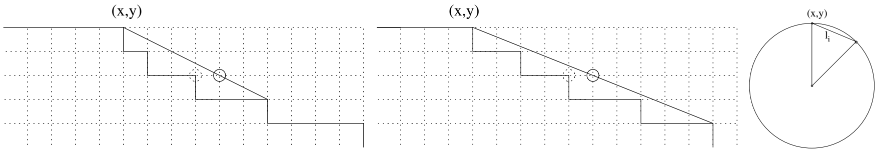


FIGURE 5.5: (Left and Centre) Showing the initial point (x, y) the chord and the point on the chord, the solid circle. (Right) Showing a chord on a circle from (x, y)

By direct construction it is easy to check that a line segment can be of the form 10^n or 10^n10^{n+1}. However we can show that no line segment may be composed of more than two chain codes which increase in length by one.

Lemma 5.10. *No line segment can be of the form* $10^n10^{n+1}10^{n+2}\ldots10^{n+i}$*, where* $i > 1$*.*

Proof. Let us first show that any line segment which is part of the discrete circle cannot be of the form $10^n10^{n+1}10^{n+2}$ for any $n > 1$. We will prove it by contradiction. Assume that the line segment $l_i = 10^n10^{n+1}10^{n+2}$ with the gradient $1/n+1$ starting at a point p with coordinates (x, y) and finishing at $(x + 3n + 6, y - 3)$. Therefore a point $(x + 2(n + 2), y - 2)$, i.e. a point which can be reached by a code 10^n10^{n+2} from a point (x, y), is above the discrete circle. By the definition of the chordal property of the circle any line that joins two points on the circle must bound, within the triangle formed from the two points on the circle and its centre, points which are again within the circle. On the other hand a point $(x + 2(n + 2), y - 2)$, shown circled in Figure. 5.5, belongs to a chord between end points of a chord (x, y) and $(x + 3n + 6, y - 3)$ and therefore should be within a discrete circle. So it is not possible to have a line segment with the following chain code $10^n10^{n+1}10^{n+2}$. The same argument holds for the general case $10^n10^{n+1}10^{n+2}\ldots10^{n+i}$, where $i > 1$ since the extension of the line segment still keep the point $(x - 2, y + 2(n + 2))$ above the discrete circle and on the other hand this point will be in the triangle formed by a centre of the circle and two end points of the line segment since its gradient is equal to $\frac{i+1}{n(i+1)+(i+1)+i(i+1)/2} = \frac{1}{n+1+i/2} \leq \frac{1}{n+2}$, where $i \geq 2$. □

Further to this weak restriction, that the chain codes may not be monotonically increasing or indeed non-decreasing, it is possible to give a more definite generalisation of the structure of the chain codes. This structure, which must be adhered to for all discrete discs, is given here.

Theorem 5.11. *Any line segments on the discrete circle with non-negative gradients should be in one of the following forms* $(10^n)^*$, $(10^n)(10^{n+1})^*$, $(10^n)^*(10^{n+1})$.

Proof. Following Lemma. 5.10 we can restrict number of cases since any concatenations of more than two chain codes which increase in length by one may not be part of a line segment. The line segments are sub-words of a chain code with increasing gradients, so if a subword $(10^n)^m$ is surrounded by any chain code segments 10^{n_1} and 10^{n_2}, where $n_1, n_2 \neq n \pm 1$, then $(10^n)^m$ is a line segment. In the case were $n_1, n_2 = n \pm 1$ a subword $(10^n)^m(10^{n+1})^k$ can be surrounded by only chain codes 10^{n_3} and 10^{n_4}, where $n_3 \neq n - 1$ and $n_4 \neq n + 2$ (by Lemma. 5.10). We will show now that the only line segments of the form $(10^n)^m(10^{n+1})^k$ can be where either $m = 1$ or $k = 1$. The proof is similar to Lemma. 5.10. Let us first show that the line segment from a point (x, y) cannot have the following chain code $(10^n)^2(10^{n+1})^2$. If we assume that such chain code may correspond to a line segment such that a point $(x - 2, y + 2n + 3)$, shown in the centre diagram in Figure. 5.5, is above the chain code (i.e. our of a circle) and at the same time is on a chord between points (x, y) and $(x - 4, y + 4n + 6)$, so should be in a discrete circle. Extending the chain code $(10^n)^2(10^{n+1})^2$ from the left by $(10^n)^{m_1}$ or from the right by $(10^{n+1})^{m_2}$ for any $m_1, m_2 > 0$ will not change the property of the point $(x - 4, y + 4n + 6)$. So it can be in one of the following forms either $(10^n)^m 10^{n+1}$ or $10^n(10^{n+1})^k$. □

Chapter 6

Pattern Formation and Composition.

Building on the categorisation of the chain codings of discrete circles, that has been presented in the previous chapter, this chapter will focus on the use of such objects to compose any number of discrete discs. This technique will allow the restrictions that are naturally imposed by the structures of the discs themselves to be relaxed. Further to this an exploration of methods of functional aggregation of differing broadcasting sequences shall be put forth, again, with an interest in removing the restrictions that are imposed on neighbourhood sequences. For these reasons results regarding the composition of discrete discs, which lead to construction of discs that would otherwise not be possible, are presented, with an algorithm for their composition given. Aggregation functions, an extension and further generalisation to the concept of nodal patterns, are used to remove the constraint of convexity on the polygons that are generated. Using this technique of aggregation experimental results pertaining to the approximation of a particular form of non-convex metric, the astroid, are given, which were previously unobtainable.

6.1 Composition

This section deals with the composition of the chain codes that have been previously characterised. The following definitions, and further characterisations, shall be required in order to discuss the properties of their composition and develop the proofs.

Definition 6.1. A discrete disc is composed of points in the $\mathbb{Z}^2$ set $\zeta^{r^2} = \{(x,y)|x^2 + y^2 \leq r^2,\ x,y \in \mathbb{Z},\ r \in \mathbb{R}\}$.

Definition 6.2. The composition of l digital discs is defined as $\zeta^{r_0} \circ \zeta^{r_1} \circ ... \circ \zeta^{r_l}$ for $r_i \in \mathbb{N}$, where, $\zeta^r \circ \zeta^{r'} = \{a + b|a \in \zeta^r,\ b \in \zeta^{r'}\}$ and is equivalent to $H(\mathcal{A}_l)$ where $\mathcal{A} = r_0, r_1, ..., r_l$ as given in Definition. 2.17 and Definition. 2.19 respectively.

Definition 6.3. The composition $u \circ v = w$ represents the composition of the chain codes in the first octant for their respective discs, ζ^u and ζ^v, which results in w the chain code for the first octant of the resultant disc of $\zeta^u \circ \zeta^v$. This method can be seen in Figure. 6.1.

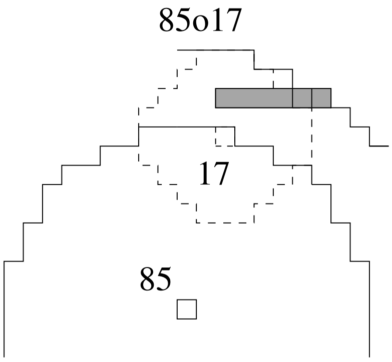


Figure 6.1: Showing the composition of squared radii $Disc(85) \circ Disc(17)$ where $Disk(85)$ is shown as a solid line and 17's first composition is shown dashed. The hatched area represents all contributions from the previous compositions.

Example 6.1. *Iterative composition of discrete circles may create different polygons. For example, in the following picture, Figure. 6.2, there are two differing cases, in one case we iteratively apply the digital circle with squared radius* 9 *resulting in the equilateral octagon. In the second case squared radii* 16 *and* 26 *are periodically applied starting with* 16.

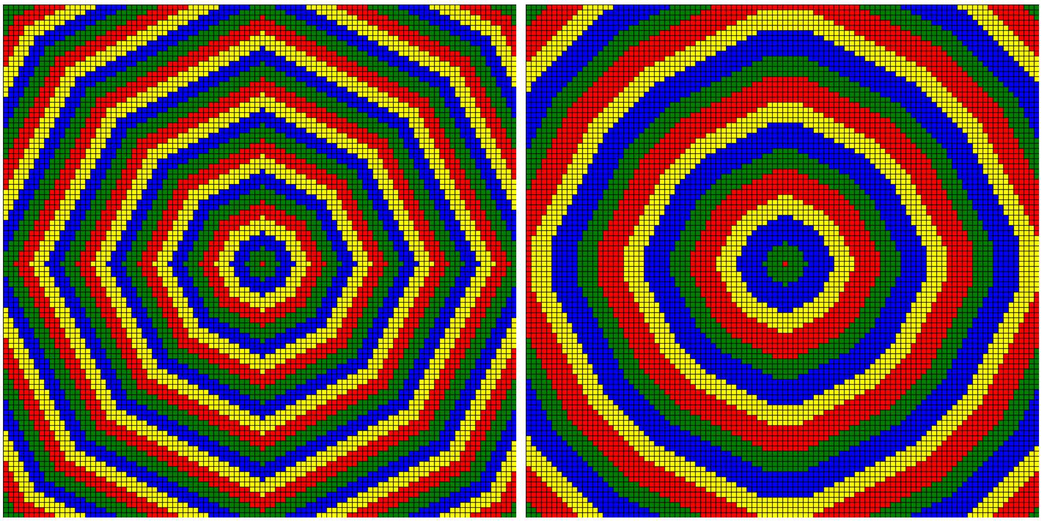


FIGURE 6.2: The above figures illustrate (left) the constant iteration of the discrete disc of squared radius 9 and (right) the alternation between the transmission of squared radii 16 then 26.

From these definitions it is now possible to describe a naive algorithm, and derive its correctness, for the composition of any number of chain codes that represent discrete discs. The result is again a convex polygon that represents the combination of the constituent parts. Such an algorithm merely systematically checks all possible combinations of the chain code and reports the largest that is found for that step. Later it shall be seen that if the chain code of the discrete disc is categorised further it is possible to give a simple linear time algorithm with a proof of correctness that is derived from the Minkowski sum along with a combinatorial proof.

Lemma 6.4. *Given two chain codes, $u = u_0u_1...u_n$ and $v = v_0v_1...v_m$, in form $\{0,1\}^*$ then $u \circ v = w = 0^{|w_0|}10^{|w_1|-1}...10^{|w_{m+n-1}|-1}$, such that*

$$|w_k| = max(\{\sum_{i=0}^{n}|u_i| + \sum_{j=0}^{k-n}|v_j| \mid 0 \leq n \leq k\}).$$

Proof. The naive way of generating the composition $u \circ v$ is for all points on the chain code of the disc $u = 0^{|u_0|}10^{|u_1|-1}...10^{|u_i|-1}...10^{|u_n|-1}$, to be the centre of the disc $v = 0^{v_0}10^{|v_1|-1}...10^{|v_j|-1}...10^{|v_m|-1}$. This is equivalent to placing the centre of the chain code v at every point defined by the chain code u generating w. The maximal point in each chain code segment (i.e if $u_i = 100$ then the coordinate of the final 0) need only be considered, clearly covering all others. Let $u_i \circ' v_j = w_{i+j}$ denote the centring v at the coordinate reached by the maximal point of u_i in which we consider the maximal point attained by chain code segment v_j and represent a possible length of the chain code for w at w_{i+j}. The maximal of all such combinations of lengths, those for which the sum of i, j are equivalent, is required and is defined as the longest contiguous subsequence of length k from $v = v_0v_1...v_j$ and $u = u_0u_1...u_i$ for all i, j such that $i+j = k$ which represents w_k for $0 \leq k < m+n$ i.e:

$$max\left(|u_0| + |v_0|\right) = |w_0|$$

$$max\begin{pmatrix} |u_0| + |v_0| + |v_1| \\ |u_0| + |u_1| + |v_0| \end{pmatrix} = |w_1|$$

$$...$$

$$max\begin{pmatrix} |u_0| + |v_0| + |v_1| + |v_2| \\ |u_0| + |u_1| + |v_0| + |v_1| \\ |u_0| + |u_1| + |u_2| + |v_0| \end{pmatrix} = |w_3|$$

$$...$$

$$max\begin{pmatrix} |u_0| + |v_0| + |v_1| + ... + |v_k| \\ |u_0| + |u_1| + |v_0| + ... + |v_{k-1}| \\ ... \\ |u_0| + |u_1| + ...|u_i| + |v_0| + ... + |v_{k-i}| \\ ... \\ |u_0| + |u_1| + ... + |u_k| + |v_0| \end{pmatrix} = |w_k|$$

$$...$$

$$|u_0| + |u_1| + ... + |u_n| + |v_0| + |v_1| + ... + |v_m| = |w_{m+n}|$$

□

Such an algorithm allows a characterisation of the composition of discrete discs. Indeed, here, it can be shown that the composition of chain codes, which have been shown to be a word $W \in \{0,1\}^*$, is, again, a word, $W' \in \{0,1\}^*$.

Corollary 6.5. *Given two chain codes u, $v \in \{0,1\}^*$. The composition $u \circ v \in \{0,1\}^*$*

Proof. Each segment in u, v, is non-zero $\forall i$ $|u_i| \neq 0, |v_i| \neq 0$. It is followed from Lemma. 6.4 that by enlarging k, for $l(k+1)$, the previous set of solutions is extended by a chain code from u or v so $l(k) < l(k+1)$. Thus the composition will be in $\{0,1\}^*$. □

Also as a derivation from the algorithm an observation about the properties of the composition function $\circ$.

Theorem 6.6. *The composition $\circ$ of two chain codes, u and v is commutative, i.e. $u \circ v = v \circ u$.*

Proof. Following Lemma. 6.4 it is possible to exchange u and v and still obtain the same result, showing commutativity of composition of chain codes u and v. The following two sets representing k-th level of a new chain code are equal as well as their maximums:

$$\{S|S = \sum_{i=0}^{n} |u_i| + \sum_{j=0}^{k-n} |v_j|, 0 \leq n \leq k\} = \{S|S = \sum_{i=0}^{n} |v_i| + \sum_{j=0}^{k-n} |u_j|, 0 \leq n \leq k\}$$

□

It is also notable that having shown commutativity chain code composition it may be possible that it is an Abelian group, where the identity element is simply the circle of radius 0, associativity is derived from the algorithms ordering of line segments by gradient, which is naturally the same irrespective of the order it is carried out in, the inverse is simply the removal of one set of line segments from a chain code. However, it is currently unknown whether or not the operation is closed.

Further, a proof of the ordering of the line segments in the discrete circle is here required in order to validate the proof of the composition theorem that is given as one of the main tools and results of this section.

Lemma 6.7. *Given line segments, l_i, of a form produced by the digitisation of a circle , which are composed of chain code segments of its first octant $a = 10^{m-1}$ and $b = 10^m$, then the ordering of their gradient for combinations of a and b is*

$$G(b) < ... < G(ab^*) < ... < G(ab) < ... < G(a^*b) < G(a) \ .$$

Proof. The proof is a simple comparison of the gradients which shows, where $|a| = m$, $\frac{1}{m+1} < \frac{n}{(n-1)(m+1)+m} < \frac{n-i}{(n-1-i)(m+1)+m} < \frac{n-i}{(n-1-i)(m)+m+1} <$
$< \frac{n}{(n-1)(m)+m+1} < \frac{n}{(n-1)m+m+1}$, where $i < n-1$, for $G(b) < G(ab^{n-1}) < G(ab^{n-1-i}) < G(a^{n-1-i}b) < G(a^{n-1}b) < G(a)$ respectively. □

Given the preceding validation about the convexity of the discrete discs it is now possible to employ the Minkowski sum to conclude and validate the composition of any number of discrete discs using the composition function.

Theorem 6.8. Composition Theorem. *Given two chain codes u and v which contain line segments $l_1^u l_2^u \dots l_t^u$ and $l_1^v l_2^v \dots l_{t'}^v$ with strictly increasing gradients. The chain code of a composition $u \circ v$ can be constructed by combining line segments of u and v and ordering them by increasing gradient.*

Proof. One of the ways to prove the above statement is to use a similar result about the Minkowski sum [51] of convex polygons in $\mathbb{R}^2$ and then to prove that it holds for the ordering of the segments of the digital circles in $\mathbb{Z}^2$. Here, a purely combinatorial proof is given in terms of chain codes, chain code segments and line segments. The above statement is proved by induction. The fact that the base case for the composition of the first two line segments holds can be seen by directly checking the expression from Lemma. 6.4.

Assume that the statement of the lemma holds and the first z line segments of $u \circ v$ were composed from a set $LS = \{l_1^u, l_2^u, \dots, l_i^u, l_1^v, l_2^v, \dots, l_j^v\}$ and contains x

chain code segments. Let us also denote a set with other line segments from u and v as $\bar{L}S$. Without loss of generality suppose that the last line segment that has been added is l_j^v, so $G(l_j^v) \geq G(l')$, where $l' \in LS$ and $G(l_j^v) \leq G(l'')$, where $l'' \in \bar{L}S$. By adding the next line segment l''' which will have a gradient larger or equal than all other line segments in LS and smaller or equal than gradients of $\bar{L}S$ it must be shown that the extended chain code of $u \circ v$ is still correct, i.e. it satisfies to Lemma. 6.4.

First of all note that adding a new part of the chain code would not change the previous x layers. If a current maximum is in the form

$$|u_p| + |u_{p-1}| + ... + |u_0| + |v_0| + |v_1| + ... + |v_q|$$

then the next $\#_1(l''')$ sums will be extended by chain codes from l'''. If l''' is a line segment in u the gradient of $G(l_j^v)$ is greater then $G(l''')$ and, taking into account Lemma. 6.7 and Theorem. 5.11, it can be seen that

$$|u_{p+1}| + |u_p| + |u_{p-1}| + ... + |u_0| + |v_0| + |v_1| + ... + |v_q|$$

is a maximum within the following set

$$\{|u_{p+1+shift}| + ... + |u_p| + |u_{p-1}| + ... + |u_0| + |v_0| + |v_1| + ... + |v_{q-shift}|, shift \in \mathbb{N}\}$$

since the shift will represent removal of a larger value from the v component and appending the smaller value from the u component.

Repeating the procedure and extending the sum by one value it can be seen that, again,

$$|u_{p+2}| + |u_{p+1}| + |u_p| + |u_{p-1}| + ... + |u_0| + |v_0| + |v_1| + ... + |v_q|$$

is a maximum within a set

$$\{|u_{p+2+shift}| + ... + |u_p| + |u_{p-1}| + ... + |u_0| + |v_0| + |v_1| + ... + |v_{q-shift}|, shift \in \mathbb{N}\}$$

following the same reasoning. Similar arguments can be applied for the case were $l''' \in v$. □

Example 6.2. *Let us illustrate the composition of two chain codes corresponding to digital circles with squared radii 45 and 9 in the first octant. The chain code of the first octant is* $\zeta^{45} = 0001$ *and for* $\zeta^{9} = 10$ *under composition this yields* $\zeta^{45} \circ \zeta^{9} = 000101$*.*

From the preceding notions it is now possible to describe a linear time algorithm, which vastly out performs the exhaustive search of all combinations that has previously been given, which is able to form the composition of the chain codes of any two discrete circles.

Algorithm: As Lemma. 6.7 shows that all of the line segments will be in a non-increasing order it is possible to construct such line segments by counting the number of $0's$ after each delimiting 1 combining those that increase in size with the preceding smaller chain code segment. Having found all line segments for both chain codes it is a simple case of, starting with the chain code with the line segment with the largest gradient, adding that element to a new chain code which is the composition of the initial two. Continue choosing the line segment with the largest gradient from either of the initial chain codes until there are no elements left in either of the original circles. The result of the algorithm is the composition of the two circles.

The following proposition gives a proof of the algorithm and the notion that it is linear time computable.

Proposition 6.9. *Two discrete circles chain codes can be composed into a single chain code in a linear time.*

Proof. As Lemma. 6.7 shows that all of the line segments will be in a non-increasing order it is possible to construct such line segments by counting the number of $0's$ after each delimiting 1 combining those that increase in size with the preceding smaller line segment. Having found all line segments for both chain codes it is a simple case of starting with the chain code with the line segment with the largest gradient adding that to a new chain code which is the composition of the initial two. □

It is also now possible to state a more formal definition of 'similar' such that it is possible to mathematically determine whether two objects are 'similar', geometrically. The following algorithm is given as one application of such an algorithm and also hints at the notion of composing one discrete disc from other discrete discs which allows an insight in to the primality of discrete discs those that cannot be composed from any other set of discrete discs.

Theorem 6.10. *Given a finite broadcasting sequence of radii $R = (r_1, r_2, ..., r_l)$ and a convex polygon P, it is decidable whether there are radii such that the chain coding of the composition of is* **similar** *to P.*

Proof. The algorithm computes all line segments, and their corresponding gradients, for the chain codes of the set of digital disks with radii in, $\mathbb{R}$, and the convex polygon, P. Each digital disk $r_i \in R$ can be represented as a vector with k values, where k is the cardinality of the set of gradients and each vector component corresponds to a particular gradient with an integer value standing for the number of line segments with this gradient in R. Finally we would need to solve a system of linear Diophantine equation over positive integers to check whether there is a set of factors for defined vectors that may match a vector for P with another unknown factor. □

6.2 r-neighbourhood Sequences and their Limitations

This section offers an analysis on limitations of the forms of circles which may be found under composition of any radii. It is from these limitations that the techniques of broadcasting sequences with aggregation and composition shall become especially fruitful in expanding the usefulness of the described techniques.

It's seen that by Lemma. 5.10 there are only three possible forms which the lines that comprise the discrete circle may take. Translating these in to their respective gradients gives the following three possible gradient forms

$$G_1 = G((10^{n-1})^m) = \frac{m}{mn} = \frac{1}{n}$$

$$G_2 = G((10^{n-1})(10^n)^m) = \frac{m+1}{n+m(n+1)}$$

$$G_3 = G((10^{n-1})^m(10^n)) = \frac{m+1}{nm+n+1} \; .$$

The above gradients are reduced to a minimal form in order to elucidate the exclusivity of the gradients.

Proposition 6.11. *It is only possible to express gradients in the first octant of a circle in a reduced form* $\frac{1}{n}$, $\frac{a}{a(n+1)-1}$ *or* $\frac{a}{an+1}$ *for* $a, n \in \mathbb{N}$.

Proof. As G_1 is already in a minimal form it is obvious that this can only express those gradients, g, of the form $\frac{1}{n}$. For G_2 and G_3 the following method is employed, $G_2 = \frac{m+1}{n+m(n+1)} = \frac{a}{b}$, where $a \perp b$ from which it follows that $m = a - 1$ such that through substitution of m in to denominator, $\frac{a}{a(n+1)-1}$ is arrived at. Similarly for G_3 by substitution the equation $\frac{a}{an+1}$ arises. □

It now becomes more clear of what cannot be expressed and the limitations inherent in the hulls of the broadcasting sequences.

Proposition 6.12. *Not all rational gradients,* $0 \leq g \leq 1$, *are expressed by the lines that comprise the discrete circle in the first octant.*

Proof. A counter example is given as proof. It is impossible to express any such rational of the form $\frac{5}{8}$. It is clear that it is not possible for G_1 to express such a fraction. For G_2 and G_3 the following suffices. $G_2 = \frac{a}{a(n+1)-1}$ where $a = 5$ such that $\frac{5}{5n+4}$ where there is no such $n \in \mathbb{N}$ such that $G_2 = \frac{5}{8}$. Similarly for $G_3 = \frac{a}{an+1}$ where $a = 5$ and $\frac{5}{5n+1}$ there is no such $n \in \mathbb{N}$ such that $G_3 = \frac{5}{8}$ □

It also becomes clear from the composition theorem (Theorem. 6.8) itself that it is not possible to generate any further gradients or new line segments through composition and in turn successive broadcasts of radii may not generate any polygons with chain codes that include gradients not previously present.

Corollary 6.13. *The set of line segments, and as such gradients, that compose any discrete circle are closed under composition.*

As noted here the shapes that are generatable are limited by their convexity as well as by the gradients of the lines that compose them. In the next section some of these limitations will be removed or relaxed using multiple broadcast sequences and an aggregation function with which to map all values to a single value.

6.3 Reducing Restrictions Through Aggregation

This section makes uses of the notion of aggregation to reduce the restrictions that are imposed by the continual composition, or simple construction of discrete discs. It can be shown that it is possible in certain cases, which shall be exposited both here and in further sections where it is employed to show the approximation of the astroid metric.

As all points may be labelled by their distance over some arbitrary modulo value an extension to the current work is proposed whereby two differing $r-neighbourhood$ sequences, A and A' are used to label the $\mathbb{Z}^2$ lattice from the same point p. At any point, p', there are now two labels such that $p' = (i, j)$ where by $k \equiv i \mod m$ for the sequence A and $k' \equiv j \mod m$ for the sequence A'. Here two differing functions for the aggregation of values of i and j which define new shapes on the lattice and in turn new metrics. The new shapes defined by the lattice are not necessarily convex.

As all of the discrete disks which make up the labellings of the lattice are composed of discrete lines it is possible to analyse the effects of the combinations of disks by considering only the intersections of the lines that make up the disks. Consider a series of parallel lines expressed in the form $y_0 = m_0x_0 + c_0$ where $c_i \in \mathbb{Z}$ is an arbitrary constant or offset, $m_i \in \mathbb{Q}$ is any gradient permitted by the line segments of a discrete circle and $x, y \in \mathbb{Z}$ are the usual Euclidean coordinates. These lines are such that each successive line is of a distance w_i from the last which shall imitate the width between iterations of the discrete circles, for the first line segment this is equivalent to $\lfloor\sqrt{r^2}\rfloor$, and the discrete lines that they generate. All coordinates such that $m_0x_0 + c_0 + kw_0 \leq y_0 < m_0x_0 + c_0 + (k+1)w_0$ are labelled as k_0. A second set of parallel lines differing from the first are defined as $y_1 = m_1x_1 + c_1$ where all coordinates such that $m_1x_1 + c_1 + k_1w_1 \leq y_1 < m_1x_1 + c_1 + (k_1+1)w_1$ are labelled k_1. The intersection of these two areas is thus (k, k'). Increasing the offset of k_0 and k_1 to $k_0 + 1$ and $k_1 + 1$ naturally results in the labelling of the area of their intersection as $(k_0 + 1, k_1 + 1)$. The ordering of the tuple is not relevant to the functions that aggregate them.

The first of the functions here have previously been studied in [54] with regards to geometric computations on the lattice. It is of particular interest in the Broadcasting Automata model where the notion of waves as observed in nature are used to control a large number of distributed automata. Being able to predict the resultant shapes that are formed by the transmission of waves is, naturally, largely advantageous in that it affords the ability to predict and manipulate the formations on the plane. Such formations can then be used for a variety of computational duties such as partitioning and geometric computation.

Waves are represented as strings of the form $(1, 0, -1, 0)$ and pairs of such values result in the nodal pattern, given in Definition. 3.1, that is resultant on the grid after distribution by broadcasting sequences. Here, summation is used over these values, much like the natural aggregation of the superposition of waves, to derive four sequences of the following form;

$$(2, 0, -2, 0) = a$$

,

$$(1, -1, -1, 1) = b$$

,

$$(0,0,0,0) = c$$

and

$$(1,-1,-1,1) = d$$

which are all pairs of cyclic permutation.

Two functions for aggregation will now be introduced with the relevant equations for resultant patterning of the lattice.

Definition 6.14. The **moiré** aggregation function is given in the table below.

moiré(x,y)=

$\oplus$	0	1	2	3
0	a	b	c	b
1	b	a	b	c
2	c	b	a	b
3	b	c	b	a

Definition 6.15. The **Anti-moiré** aggregation function can be expressed simply as addition over modulo 4 and is given in the table below.

Anti-moiré(x,y)=

$\oplus$	0	1	2	3
0	a	b	c	d
1	b	c	d	a
2	c	d	a	b
3	d	a	b	c

Proposition 6.16. *The gradients of the new lines formed by the moiré equation can be predicted using the equation*

$$\frac{w_o \cdot m_1 - w_1 \cdot m_0}{w_0 - w_1}.$$

Variables refer to the, line gradient, m_i, and the width of the line, w_i.

Proof. The form of the moiré function is such that if the intersection of one area is labelled as (k, k') then the next area that is labelled similarly will be of the form $(k+1, k'+1)$. Encoding this as the intersections of lines $y = m_0x + c_0 + k \cdot w_0$ with $y = m_1x + c_1 + k' \cdot w_1$ and $y = m_0x + c_0 + (k+1) \cdot w_0$ with $y = m_1x + c_1 + (k'+1) \cdot w_1$. All that is left is to find the gradient of the two intersections. Solving the first case, $x_0 = \frac{c_1 - c_0 - k \cdot w_0 - k' \cdot w_1}{m_0 - m_1}$ and $y_0 = m_0(\frac{c_1 - c_0 - k \cdot w_0 - k' \cdot w_1}{m_0 - m_1}) + c_0 + k \cdot w_0$. The second, $x_1 = \frac{c_1 - c_0 - (k+1) \cdot w_0 - (k'+1) \cdot w_1}{m_0 - m_1}$ and $y_1 = m_0(\frac{c_1 - c_0 - (k+1) \cdot w_0 - (k'+1) \cdot w_1}{m_0 - m_1}) + c_0 + (k+1) \cdot w_0$. The gradient of the two points can be found, $\frac{y_1 - y_0}{x_1 - x_0}$, which results in the gradient of the labelling applied to the lattice. □

Proposition 6.17. *The gradients of the new lines formed by the Anti-moiré equation can be predicted using the equation*

$$\frac{w_o \cdot m_1 + w_1 \cdot m_0}{w_0 + w_1} .$$

Variables refer to, line gradient, m_i, and the width of the line, w_i.

Proof. The form of the moiré function is such that if the intersection of one area is labelled as (k, k') then the next area that is labelled similarly will be of the form $(k+1, k'+1)$. Encoding this as the intersections of lines $y = m_0x + c_0 + k \cdot w_0$ with $y = m_1x + c_1 + (k'+1) \cdot w_1$ and $y = m_0x + c_0 + (k+1) \cdot w_0$ with $y = m_1x + c_1 + k' \cdot w_1$. All that is left is to find the gradient of the two intersections. Solving the first case, $x_0 = \frac{c_1 - c_0 - k \cdot w_0 - (k'+1) \cdot w_1}{m_0 - m_1}$ and $y_0 = m_0(\frac{c_1 - c_0 - k \cdot w_0 - (k'+1) \cdot w_1}{m_0 - m_1}) + c_0 + k \cdot w_0$. The second, $x_1 = \frac{c_1 - c_0 - (k+1) \cdot w_0 - k' \cdot w_1}{m_0 - m_1}$ and $y_1 = m_0(\frac{c_1 - c_0 - (k+1) \cdot w_0 - k' \cdot w_1}{m_0 - m_1}) + c_0 + (k+1) \cdot w_0$. The gradient of the two points can be found, $\frac{y_1 - y_0}{x_1 - x_0}$, which results in the gradient of the labelling applied to the lattice. □

Composition of this form is given as example in Figure. 6.3. Here the two lines are shown on the lattice correspond to the gradients of the two differing aggregation functions.

The following proposition is now noted.

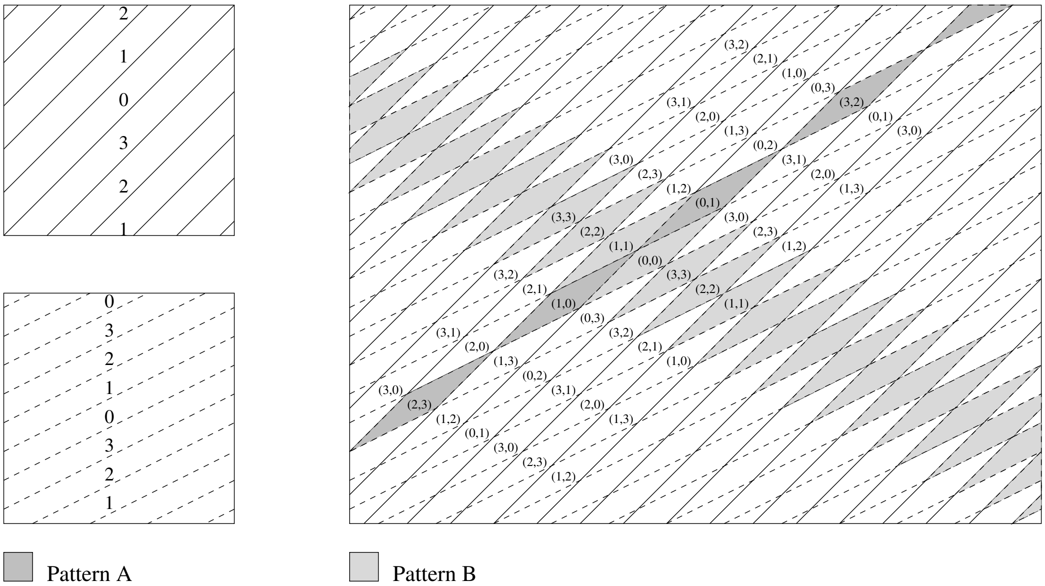


Figure 6.3: The two forms of aggregation via the above functions. Here pattern A gives the line formed by the anti-moire function and the pattern b gives the line as formed by the moire function.

Proposition 6.18. *For any two discrete lines on the $\mathbb{Z}^2$ lattice it is possible to vary the observed gradients for resultant from aggregate functions by varying the width of the lines.*

Proof. The proof is an observation of the equations for the gradients of the lines resultant from aggregation. Observing the results of two aggregating functions, $\frac{w_o \cdot m_1 + w_1 \cdot m_0}{w_0 + w_1}$ and $\frac{w_o \cdot m_1 - w_1 \cdot m_0}{w_0 - w_1}$, any alteration to the width of the line which represents results in a direct change in the gradient. □

It is now possible to formulate the following statement which shows the increase in expressivity that comes from using the composition of two broadcast sequences.

The number of lines formed by aggregation are not limited by the restrictions that are demonstrated in Proposition. 6.11 where it is now possible to construct new lines through the varying of widths and indeed new polygons may now be formed with this technique such that they are non-convex.

6.4 Formation of Polygons Through Aggregation.

Having shown that it is possible to construct different gradients from line sections it is natural to now observe what happens when whole circles are intersected and the relationships that are formed by the aggregation functions that have been introduced. It is noted that a diagram for the formation of polygons in case of the moiré aggregation function is shown in Figure. 6.4. Two digital discs, those with squared radii two and 25 such that the discs are ζ^2 and ζ^{25}, are composed generating, from the two convex polygons, and new, previously unreachable, polygon which is $non - convex$.

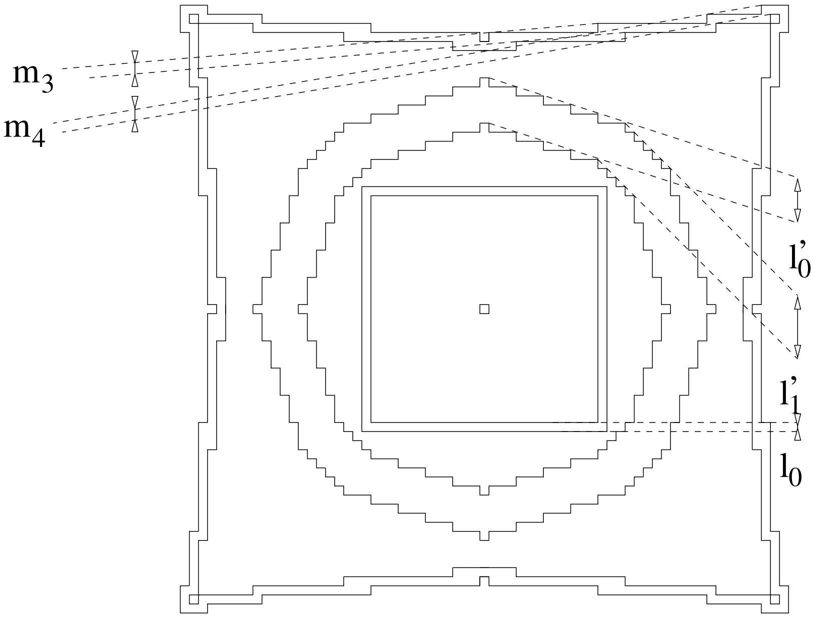


FIGURE 6.4: The above schematic depicts the broadcast of two discrete circles. The first, inner, circle (of squared radius two where $\zeta^2 = l_0$ with gradient m_0) and the second discrete circle (of squared radius 25 where $\zeta^{25} = l'_0 l'_1$ with gradients m_1 and m_2 respectively) the outer construction shows the resulting moiré lines here labelled m_3 and m_4. Arrowed lines show line width measurements, the respective w_i, here, $w_0 = 1$, $w_1 = 5$, $w_2 = 7$, $w_3 = \frac{5}{4}$ and $w_4 = \frac{7}{6}$.

The following examples given in Figure. 6.5 also elucidate the differences between the two aggregation functions showing the non-convex polygon generated by the two digital discs that are represented by, ζ^2 and ζ^9.

It is also possible to gain a better picture of the overall shapes produced by the anti moire equation by reducing the number of labels, and so the colours, further.

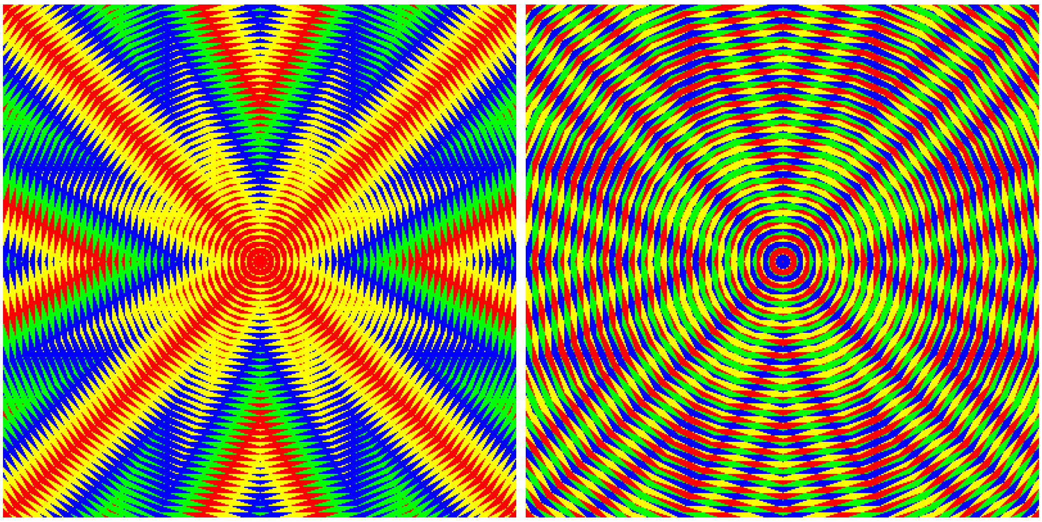


FIGURE 6.5: Above gives an example of aggregation of two broadcast sequences, one of the discrete disc ζ^{32} and the other ζ^{36}, from a central point on the diagram and with the two aggregation functions (left) moiré and (right) Anti-moiré.

This is done by a process of merging certain values or reducing the values over some modulo which in this example, Figure. 6.6, is two. The reduction is given by the following aggregation function:

Anti-moiré-Mod2(x,y)=

$\oplus$	0	1	2	3
0	a	b	b	a
1	b	b	a	a
2	b	a	a	b
3	a	a	b	b

6.5 Approximating L^p Metrics with Composed Broadcasting Automata

Previously, von Neumann and Moore neighbourhoods have been used to achieve an approximation of the Euclidean metric through some mixing of the neighbourhoods. This has been done in many different settings such as periodic, non-periodic combinations of the two neighbourhoods, regular and non-regular, i.e, hexagonal, triangular grids to which different sequences, i.e applying different definitions of

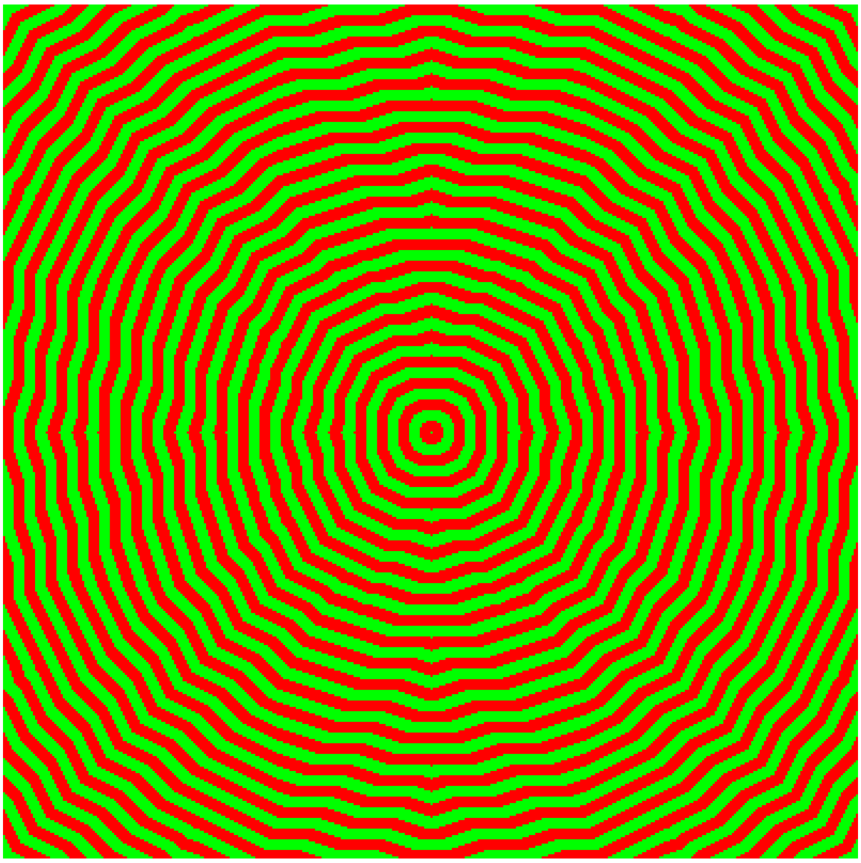


FIGURE 6.6: The above figure depicts the anti moire pattern after the merging of two sets of two colours and initially generated by the discrete discs of squared radius 32 and 36 from the central point on the plane.

neighbourhood, are applied, as well as a variety of methods for defining just how an approximation of Euclidean distance by neighbourhood sequences should be defined and measured, where most of these techniques discuss notions of digital circularity such as the isoperimetric ratio, perimeter comparisons, etc. In short this has been one of the main studies with regards to neighbourhood sequences. There is ultimately, as has been previously discussed, a large barrier to the extension of this body of work in the approximation of the more general L^p metrics due to the impossibility constructing any non-convex polygon from the composition of the two convex polygons which represent the Moore and von Neumann neighbourhoods. The astroid is part of the 'family' of L^p metrics of which the Moore neighbourhood, L^∞, the von Neumann neighbourhood, L^1, and the Euclidean metric, L^2 are the most well known. In general an L^p space is defined by the formula $\| x \|_p = (|x_1|^p + |x_2|^p + ... + |x_n|^p)^{\frac{1}{p}}$ such that $\| x \|_p$ is the p-norm.

It is in this section that a new method for the generalisation of metric approximation with broadcasting neighbourhoods shall be given and, using only two broadcasting radii, here given in terms of r^2, and the moiré aggregating function, explore the ability of this model to approximate the astroid and, as such, a new class of metrics outside of the reach of neighbourhood sequences.

It has previously been seen, in this chapter, that the methodology of combining two broadcasting sequences using an aggregating function can be used to extend the possible resulting polygons. In this section it shall be seen that this can be employed in the solution to a practical problem. The problem, in this case, is the approximation of L^p metric, $L^{2/3}$, which forms an astroid [76]. The astroid can be expressed by the equation, $x^{2/3} + y^{2/3} = r^{2/3}$, where x and y are those of the Cartesian plane and, as with the equation for the circle from which this equation is generalised, the r is the 'radius' of the astroid.

A few preliminaries with regards to the astroid which are taken from [76] will be required to understand the methods proposed for comparing the approximation, via broadcasting sequence and aggregation, and the actual astroid. The astroid was discovered in Roemer in 1674 whilst searching for the best form of gear teeth. It is a hypercycloid of four cups and can be described by a point on a circle of radius $\frac{3}{4} \cdot a$ rolling on the inside of a fixed circle of radius a. The resultant shape has a perimeter, L, of $L = 6a$ where a is the radius of the outer fixed circle and is the maximal point reached by the astroid. The area, A, encapsulated is given by $A = \frac{3}{8}\pi a^2$. Finally, the point at which the x and y coordinates are equivalent on the perimeter of the astroid can be expressed as $\frac{r}{2}$. The equation is known to have applications in magnetism where the Stoner-Wohlfarth astroid curve separates regions with two minima of free energy density from those with only one energy minimum and is a geometric representation of the model of the same name [77].

In [40, 78] a number of methods for how well a particular neighbourhood sequence approximates the euclidean distance are given. The authors of [40] consider any sequence, including those sequences which are non-periodic, of neighbourhoods with which to approximate the euclidean distance. In order to show how well, or

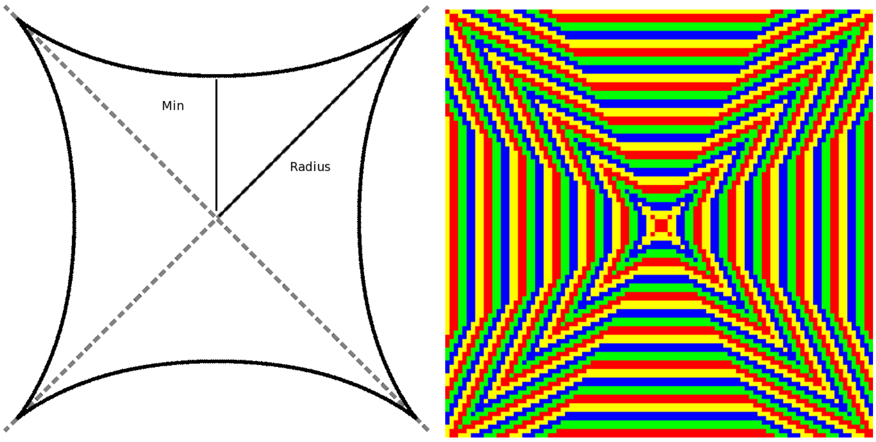


Figure 6.7: The above figure depicts the astroid, here, rotated by 45° (left) compared to its best, found, approximation with broadcasting sequences (right), an aggregation of the discrete discs of squared radius, 2 and 5.

poorly, the sequence approximates the euclidean distance it is compared to the euclidean circle through a variety of methods. One such method is through the use of the isoperimetric ratio or the noncompactness ratio. Here, the attempt is to measure $\frac{P^2}{A}$ where, P, is the perimeter, and, A, is the area. This measure is conjectured to be minimal for the circle where it is 4π, however, this does not help when looking at non-convex shapes, such as those produced when approximating the an astroid, due to the measure being optimised for convex and symmetrical shapes.

Other ways of approximating the euclidean circle are suggested such as a perimeter based approximation and area based approximation. With respects to are based approximation there are two techniques used. The first is that of the inscribed circle based approximation. This method attempts to find a sequence that generates the polygon, generated by the neighbourhood sequence, which is closest to the polygon having the given circle as the inscribed circle. The second of these methods is the covering circle based approximation such that polygon must be covered by the circle. More methods are given but rely on properties distinct to the circle and so are not discussed here.

Further, in [78], another measure of circularity is considered using three differing methods. They formulate three approximation problems whereby the aim is to find the neighbourhood sequence that minimises the error in each formulation.

The problems may be informally described as the following: problem one requires finding the neighbourhood sequence that best minimises the size of the symmetric difference between some neighbourhood sequence at step k, given as A_k and the Euclidean circle of radius k; the second problem attempts to find the sequence that best minimises the complement of the neighbourhood generated by A_k with it's smallest inscribed circle; the third problem is a discretisation of the second problem where an A_k must be found such that it minimises the complement with the largest discrete disc that can be inscribed within.

It is the second of these problems from [78] that shall be explored as a method for showing a best approximation of the astroid with combined broadcasting sequences. In this case the method is the most simplistic to calculate and also the least dependent on any of the properties that are only present in the circle. As such a more formal representation of the problem can now be posed, here, given as, $Area(H(f_k(A,B))\backslash G'_k \leq Area(H(f_{k'}(A,B'))\backslash G'_{k'})$, where, $f_k(A,B)$ is the kth polygon generated by the aggregation of the broadcasting sequences, A, which here will be the discrete disc, $r^2 = 2$, and the broadcasting sequence, B, B', here a variable which is to be optimised to find the best approximation. There is a simple change to G_k, originally used in [78] to represent the circle of radius k, to convert it to G'_k in that $G'_k = \{q \in \mathbb{R}^2 : L^{\frac{2}{3}}(0,q) \leq k\}$ as simple conversion of the metric from that of the euclidean distance, L^2, to the astroid distance, $L^{\frac{2}{3}}$.

The complexity of calculating these compositions for the purposes of optimisation means that only experimental data shall be given here as a proof of validity of the approximation of the concave metrics, which the astroid represents. As the astroids require a rotation by 45° in order to match the polygon that is generated by the aggregation, $f(A,B)$, as defined before. In matching the point that is at the largest euclidean distance from the origin, which for simplicity, and without loss of generality, is considered the initial point from which all broadcasting occurs. This point is matched to the same point on some polygon on some composition, $f(A,B)$, and the complement of the areas compared. The following table is produced with this method.

Astroid - $L^{\frac{2}{3}}$			moiré - $f_k(A,B)$					
Radius	Min	Area	Radius	Min	Area	B	k	Complement
17	8.5	340	17	11	461	5	5	121
17	8.5	340	17	13	753	9	8	413
17	8.5	340	17	13	873	10	9	533
18	9	382	18	16	1121	13	11	739
18	9	382	18	14	1033	16	11	651
18	9	382	18	14	1001	17	11	619
17	8.5	340	17	15	1041	37	13	701
17	8.5	340	17	16	1141	45	14	801
17	8.5	340	17	16	1093	61	14	753
17	8.5	340	17	16	1181	82	15	841

Table 6.1: Illustrating the experiments done with regards to the approximation of the astroid.

The function of aggregation must also be defined. Here the choice is moiré aggregation without the modulo restriction, although, such a restriction is retained in the images for simplicity. Such a function can now be defined as, $f(A_i, B_j) = |i-j|$ for the ith and jth iteration of the sequence A and B respectively. For the function $f_k(A, B)$ where there exists some A_i and B_j such that $|i-j| = k$.

The min point for the $L^{\frac{2}{3}}$ is calculated by $\frac{r}{2}$ where r is the radius of the astroid, the r in $x^{\frac{2}{3}} + y^{\frac{2}{3}} = r^{\frac{2}{3}}$. The B is the second broadcasting sequence, here, given as the r^2 of the disc. Images for each of the resultant, aggregated images are given in Table. 6.2 and Table. 6.3.

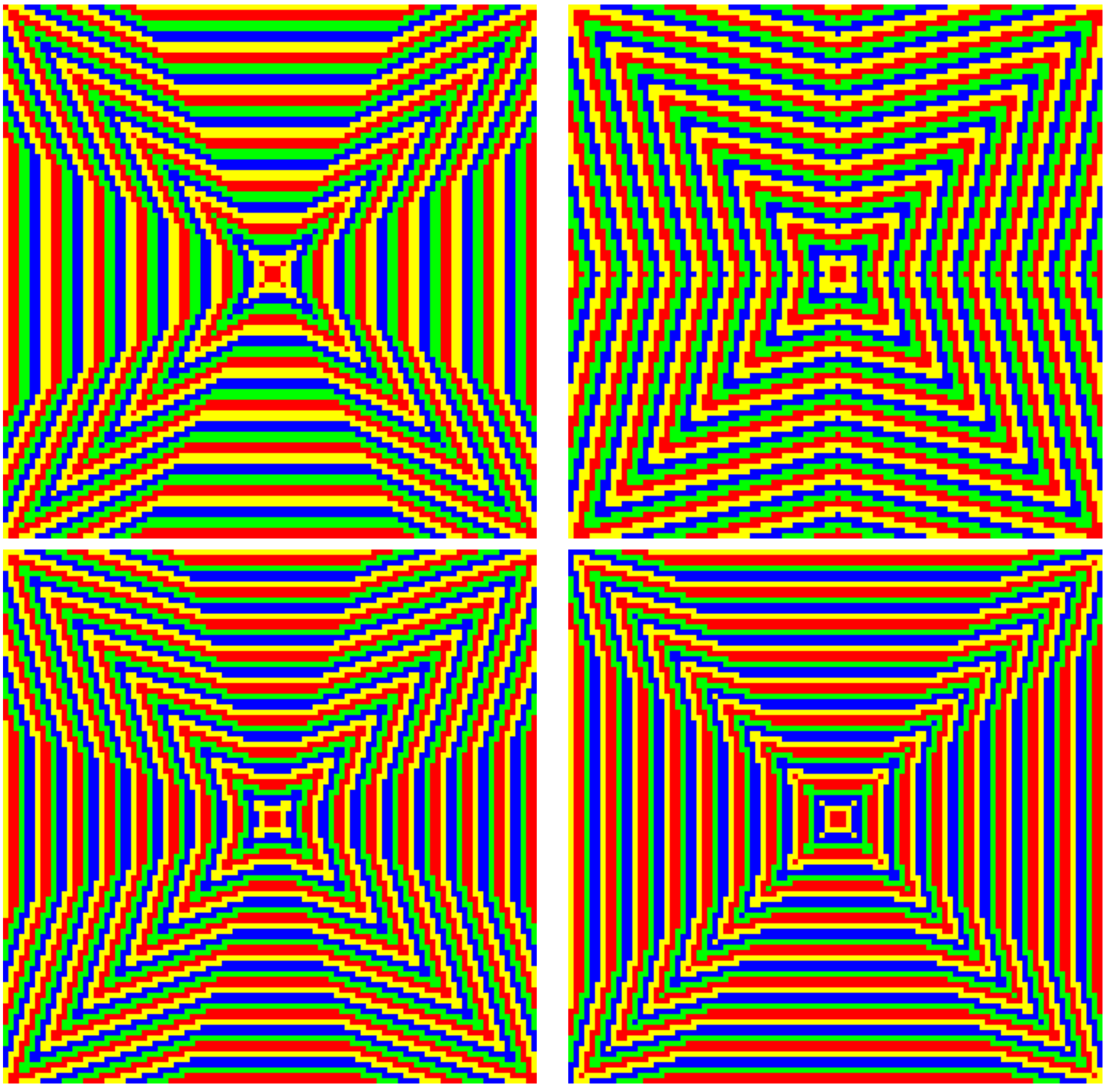


Table 6.2: The above diagrams show the patterns generated by the aggregation of the two discs, where one is the disc of squared radius 2 and the other is varied, in these diagrams (from left to right and line by line) there are discs of squared radius, 5, 9, 10 and 13.

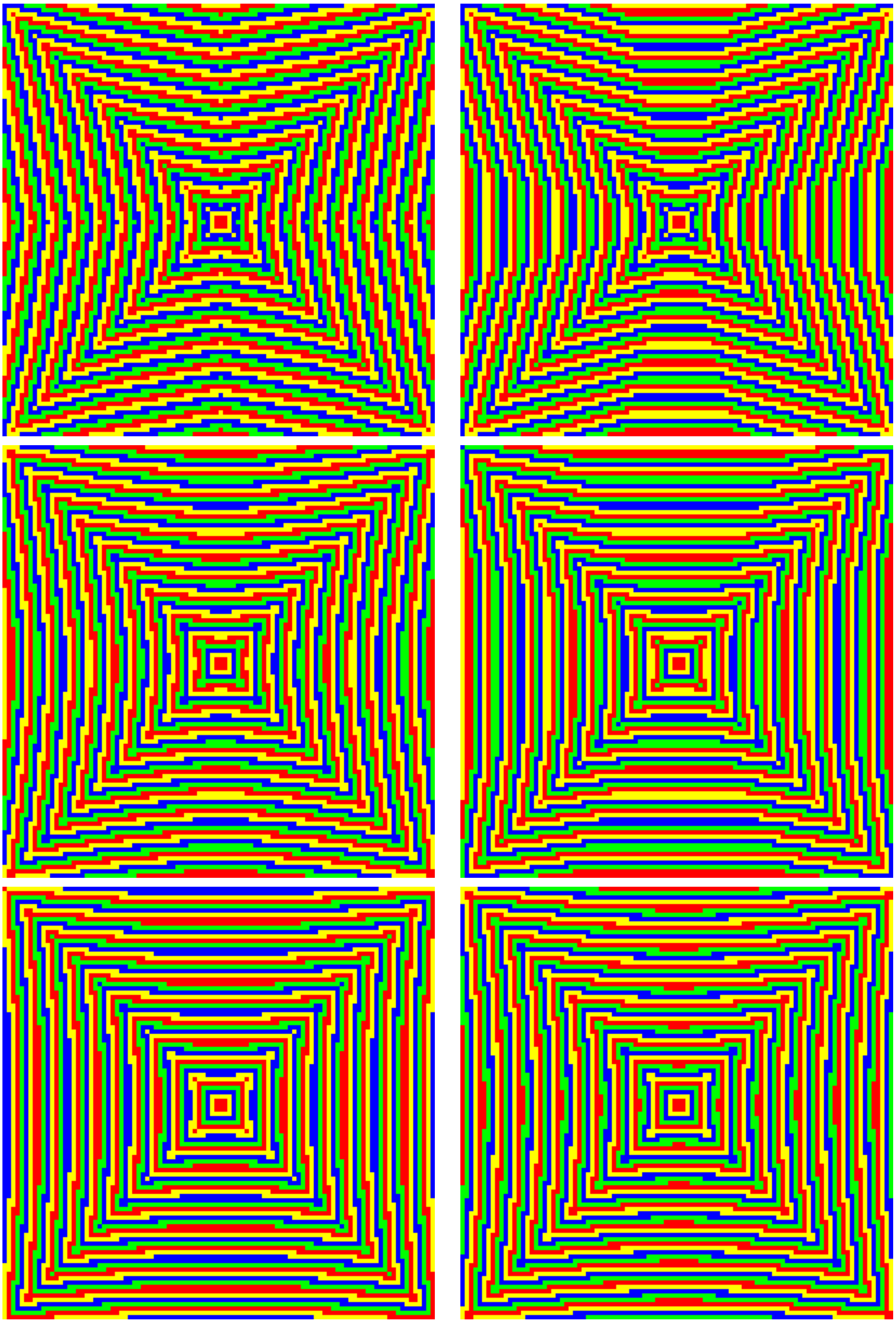


TABLE 6.3: The above diagrams show the patterns generated by the aggregation of the two discs, where one is the disc of squared radius 2 and the other is varied, in these diagrams (from left to right and line by line) there are discs of squared radius, 16, 17, 37, 45, 61 and 82.

Astroid - $L^{\frac{2}{3}}$			moiré - $f_k(A,B)$					
Radius	Min	Area	Radius	Min	Area	B	k	Complement
2	1	5	2	1	13	5	1	8
5	2.5	30	5	3	65	5	2	35
8	4	75	8	5	157	5	3	82
11	5.59	143	11	7	289	5	4	146
18	9	382	18	9	461	5	5	79
21	10.5	520	21	11	673	5	6	153
24	12	679	24	13	925	5	7	246
27	13.5	859	27	15	1217	5	8	358
30	15	1060	30	17	1549	5	9	489
33	16.5	1283	33	19	1921	5	10	628

Table 6.4: Illustrating the experiments done with regards to the approximation of the astroid where both A and B are fixed.

Table. 6.4 gives a series of comparisons for $Area(H(f_k(A,B)))\backslash G'_k$. From this table it is possible to observe that the best approximation, from those constructed, though there is a trend towards worsening approximations as B increases, that the simplest composition yields the best results. In this case this value for B is the disc generated by the squared radius r^2 of 5, this disc being constructed by the next smallest squared radius that constructs a new discrete disc. The following table now looks, again, experimentally, at how the approximation changes as k increases. The table notes that the approximation weakens as it increases perhaps indicating a divergence between the astroid and the polygon generated by $f_K(A,B)$.

6.6 Pattern Formations and Periodic Structures in $\mathbb{Z}^2$

It is natural after observing the variety of effects that are the result of the application of an aggregation function to look at functions which are themselves shapes of some form. Here, functions of this form and their resultant patterns, imposed on the grid according to their application, are given. Such functions are only restricted by their symmetry, a result of the unordered nature of the tuples to be aggregated.

The first function, depicted in Figure. 6.8 (Left) takes the form of a discrete disc itself, in this case one which is also represented by the discrete disc of squared radius five. The functions here have also been increased in size and the size of the modulo for the labels has been increased. The use of discrete discs of squared radius 8 is important here as it constructs, in some sections of the lattice, a perfect reproduction of the shape given in the aggregating function. The following table describes the aggregating function and the details of the image.

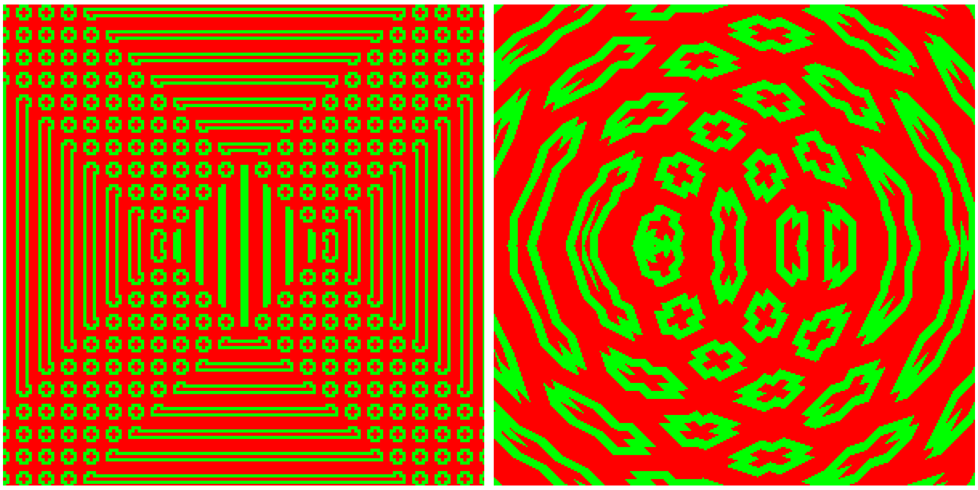


FIGURE 6.8: (Left) Patterns generated by the aggregating function representing the discrete disc of squared radius five and the labelling of the lattice given by two broadcasting sequences of squared radius eight. (Right) Patterns generated by the aggregating function representing the discrete disc of squared radius five and the labelling of the lattice given by two broadcasting sequences, one of squared radius 26 and the other of squared radius 36.

Array Size: 300

Centre 1: (100,150)

Centre 2: (200,150)

Radius 1: 8

Radius 2: 8

Modular Labelling: (0,1,2,3,4,5,6)

⊕	0	1	2	3	4	5	6
0	a	a	a	a	a	a	a
1	a	a	b	b	b	a	a
2	a	b	b	a	b	b	a
3	a	b	a	a	a	b	a
4	a	b	b	a	b	b	a
5	a	a	b	b	b	a	a
6	a	a	a	a	a	a	a

Aggregation Function: Shown right.

Figure. 6.8 (Left)

This reproduction of the aggregating function is not always exact. It is possible to skew and deform the representation of the image described by altering the radii of the circles that are used to form the underlying labelling of the lattice. The following image, Figure. 6.8 (right) gives an example of such a deformation.

Array Size: 300

Centre 1: (100,150)

Centre 2: (200,150)

Radius 1: 26

Radius 2: 36

Modular Labelling: (0,1,2,3,4,5,6)

⊕	0	1	2	3	4	5	6
0	a	a	a	a	a	a	a
1	a	a	b	b	b	a	a
2	a	b	b	a	b	b	a
3	a	b	a	a	a	b	a
4	a	b	b	a	b	b	a
5	a	a	b	b	b	a	a
6	a	a	a	a	a	a	a

Aggregation Function: Shown right.

Figure. 6.8 (Right)

The two following figures, Figure. 6.9, demonstrates changes that occur when manipulating the number of colours, changing the aggregation function and altering the underlying tuples that generate the overall shape of the colourings.

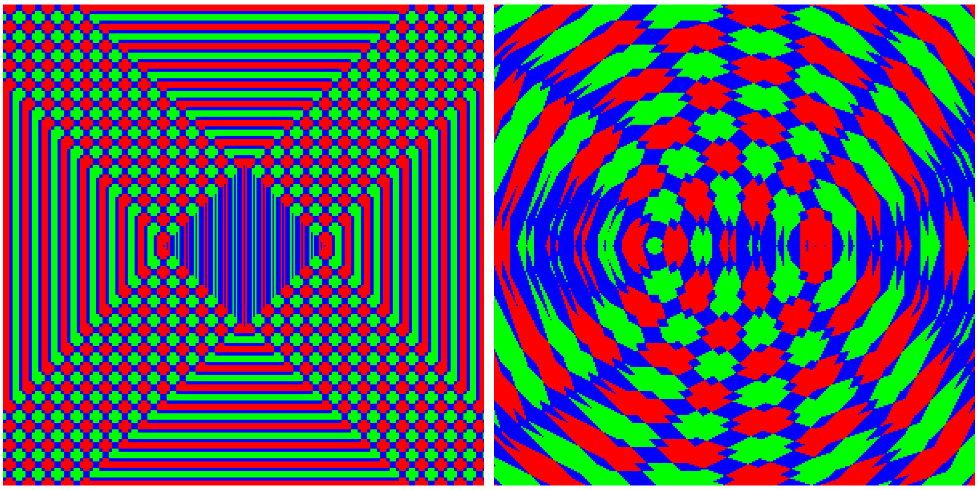


FIGURE 6.9: Patterns generated by the aggregating function represented by the table and the labelling of the lattice given by two broadcasting sequences, (right) both of squared radius eight (left) one of squared radius 26 and the other of squared radius 36.

Array Size: 300

Centre 1: (100,150)

Centre 2: (200,150)

Radius 1: 8

Radius 2: 8

Modular Labelling: (0,1,2,3,4,5)

Aggregation Function: Shown right.

Figure. 6.9

⊕	0	1	2	3	4	5
0	b	b	c	c	b	b
1	b	c	a	a	c	b
2	c	a	a	a	a	c
3	c	a	a	a	a	c
4	b	c	a	a	c	b
5	b	b	c	c	b	b

Array Size: 300

Centre 1: (100,150)

Centre 2: (200,150)

Radius 1: 26

Radius 2: 36

Modular Labelling: (0,1,2,3,4,5)

Aggregation Function: Shown right.

$\oplus$	0	1	2	3	4	5
0	b	b	c	c	b	b
1	b	c	a	a	c	b
2	c	a	a	a	a	c
3	c	a	a	a	a	c
4	b	c	a	a	c	b
5	b	b	c	c	b	b

Figure. 6.9

Altering the aggregating function is clearly a powerful tool in pattern and polygon formation. The alteration of the discs that are used as the basis of the aggregation also show that any underlying shape generate by an aggregating function can be skewed and otherwise altered whilst retaining the gestalt representation. Methods of manipulating the hew and scale of the shapes that are generated through some aggregating function may be useful to pattern recognition and detection methods that are part of the Swarm Robotics cannon among others. Whilst altering the aggregation function is one an interesting concept, for the purpose of pattern formation, there is still a lot of possibility that remains when only considering the standard moiré function. Such as can be seen in the variety of patterns that are formed in Figure. 6.10.

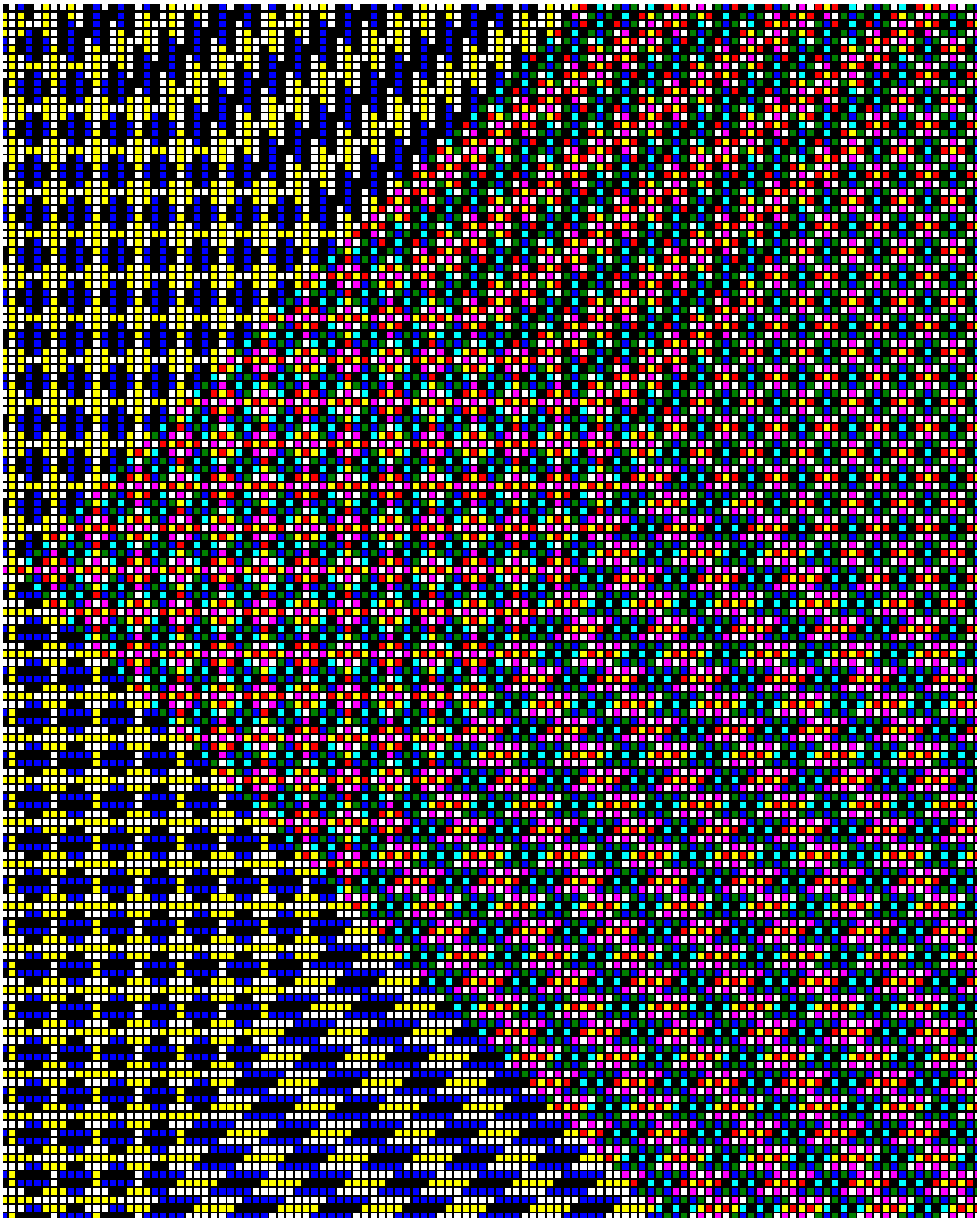


FIGURE 6.10: The above figure shows a number of distinct patterns that are here generated by four different transmissions where two, one of squared radius 2 and the other of squared radius 12, are placed at two separate points.

Chapter 7

Conclusions

7.1 New Results

The thesis begins with the statement of a new and compelling model for geometric, distributed computation as it pertains to the field of Swarm Robotics. The work primarily concerned itself, initially, with the exposition of this model, Broadcasting Automata, its possibilities and uses, as given by a series of algorithms, as derived from the natural phenomena that inspired it. In the introduction the connection to the natural phenomena are made, highlighting the elements of the physical world that are drawn on for inspiration as well as the way in which this correlates with Cellular Automata with which there is a strong comparison though, as has been seem, no such equivalence. Also, the computational aspects that are drawn on for the model are noted, with concepts taken from a variety of fields; algorithmic, distributed, ad-hoc networking and neighbourhood sequences. This is coupled with mathematical concepts, or formalisations of physical notions, such as moiré patterns, superposition, discrete circle, convex polygons and Minkowski sums.

The model then aims to represent all of these constituent parts, which, once combined are used to define a notion of distributed geometric computation. The systems in defined rigorously and in a general way such that, whilst one form of this model is completely examined here, there are many more forms and generalisations that can be made. It is here that not only the general foundation for the body of work but the first exposition of the foundations of the specific model in question is laid. The focus here is on the square lattice and which is retained throughout the rest of the thesis.

The third chapter introduces the notion of waves more concretely as being made of words. Combinations of words are defined to be nodal patterns and a series of results are given about them. These include the most important to the thesis, the situation in which there are two transmitters and two words, both the same, with a result given pertaining to the exact number of distinct combinations. Then, distance functions are given for the two primitive broadcasting radii, Moore and von Neumann, and is used to show how the previously counted nodal patterns will fall on the plane. This leads to a formulation of the patterns resultant from two sources and an indication of the ability of the automata to recognise the partitions that are formed by the nodal patterning. These results are given for two dimensions and with one and two different sources.

The fourth chapter highlights the viability and uses of the Broadcasting Automata model, elucidating a series of algorithms, a translation from a Cellular Automata algorithm in the form of the Firing Squad Synchronisation Problem, methods for locating the centre of a circle from any point on its edge which is of interest both to the fields of distributed and geometric computation as well as Swarm Robotics. These are also supported by a series of 'helper' algorithms designed to embellish the basic principles already in force and remove some of the limitations in the model that may be considered reusable or necessary in many other algorithms. The main result of this chapter is the algorithm for the location of the centre of the digital disc where the initial transmitter is located on the edge of the disc. The algorithm uses all of the techniques discussed in the previous chapter, that utilise nodal patterns, to locate the centre of the disc through distributed, geometric computations.

The variation of the broadcasting radius, which generates a discrete disc, provides a great many resultant shapes which must be studied in order to fully understand the Broadcasting Automata model. With this study required, and in mind, the fifth chapter begins with the analysis of the discrete disc. After the prerequisite notions of chain coding are dealt with, where such results are considered important with respects to the number of automata that are encompassed by the discrete disc and that the resulting shape affords control of such aspects as the resultant patterning on the plane, the angles and number of partitions and the shape of the polygon that is formed.

The final set of results pertain to the ability to compose the discrete discs giving an algorithm that is linear in time, after a further series of categorisation of the discrete disc that leads to the proof of this notion. The set of gradients that it is possible to construct with the set of natural discrete discs is given and shown to be incomplete and closed under composition.

Given the limitations of neighbourhood sequences it is shown that it is interesting, that is, that it is possible, to use the notion of aggregation to remove such perceived restrictions by allowing the generation of a series of new angles. These new angles are shown to allow the construction of convex polygons which were clearly impossible before and formula and results pertaining to the two important aggregation function moiré and anti-moiré. This notion is then used to develop a construction which is able to approximate the convex metric, the astroid, where metric approximation was one of the main sections of research in the area of neighbourhood sequences.

Finally a brief experimental exploration of the possibilities of the variety of resultant patterns that can be produced by the infinite combinations of discrete disc size, placement and composition can afford when paired with any of the, also infinite, aggregating functions.

7.2 Considerations for Further Work

As a large part of the body of this work is dedicated to foundations of a model that is only partially explored here in detail there is a great deal of work left open. As noted in Chapter 2, which discusses the model, the model can be extended to any of the common or uncommon lattices of which, for two common lattices, the hexagonal and triangular lattices, some work, in the form of neighbourhood sequences and the approximation of Euclidean metrics within those various models [40, 79, 80]. Such work means that this would be a viable extension to the Broadcasting Automata model for pattern formation and partitioning considerations and it may even suggest that the aggregation of such neighbourhood sequences may improve the approximations that have already been obtained for Euclidean distance or even, as has been shown in this thesis, to allow the approximation of metrics that weren't possible before.

Further to the exploration of different grids in two dimensions is the exploration of the square grid, as can be seen elucidated throughout this thesis, is the exploration of the square grid in three dimensions and what this means to the patterns and algorithms that have been demonstrated here, to some degree this has been considered in neighbourhood sequences [36] and so could be considered a viable extension. Naturally such considerations could also be extended to three dimensional interpretations of the various lattices that may be formed in the three dimensional Euclidean space. Exploration of these ideas would lead to new categorisations of the shapes defined by broadcast of various radii and the definition of composition that may be formed in those models. Further to the expansion of the various forms of lattices and transmission radii it is also possible to explore that model on graphs which may be of any form. Complications of this approach will most likely stem from Broadcasting Automata's exploitation of the underlying arrangement of the structure that messages will be passed through. Graphs, especially those that are random, are not necessarily constructed in such a way that it may be exploited to from resultant geometric forms from the embedding in the plane. All such models may be used to explore questions of metrics and approximations and indeed such questions will all add to the robustness of the Broadcasting Automata model.

Whilst the structure that underpins the organisation of Broadcasting Automata is, as has been exposed in this thesis, central to the concept and uses of the model, as important is the notion of aggregation of the messages that are broadcast throughout the system. Whilst a brief inspection of the addition function has been discussed here giving concrete combinatorial results on the number of distinct combination this area, is again, vast. There are many more problems that could be assessed here with respects to exact bounds for a variety of functions, words and number of transmitters. As previously suggested some such general considerations within this area can be seen in work by the name of additive combinatorics[68].

With the preceding in mind a combination of the two, analysing the resulting partitions that are formed by the various discrete metrics imposed by the transmissions and the resultant patterns formed within those partitions would also be something worth considering.

Aside from the exploration of the many different permutations of the model there are the considerations of the applications. The model naturally lends itself to the construction of fast, parallel algorithms that are attempting to tackle geometric problems. In this nature problems such as the convex hull problem, where the smallest convex set that encloses some collection of points, where, perhaps the points could be pre-elected transmitter. Another problem is that of constructing a Voronoi diagram where the aim is to construct the corresponding Voronoi cell for each point such that the set of all points in the given space whose distance to the given object is not greater than their distance to the other objects. Both such algorithms would also be useful addition to the algorithms with regards to their usefulness in the field of Swarm Robotics.

By the very nature of the project, in that it has presented a model in a new light, with aims to examine its plausibility and efficacy, which have been detailed, the scope for further work is unbounded. In this way it is only the hopes of this thesis to begin an elucidation of the possibilities that are presented by Broadcasting

Automata and it is hoped that it will be much help to the further exposition of the concepts presented here.

Bibliography

[1] F. Mondada, G. C. Pettinaro, A. Guignard, I.W. Kwee, D. Floreano, J.L. Deneubourg, S. Nolfi, L.M. Gambardella, and M. Dorigo. Swarm-bot: A new distributed robotic concept. *Auton. Robots*, 17:193–221, September 2004. ISSN 0929-5593. doi: 10.1023/B:AURO.0000033972.50769.1c. URL http://dl.acm.org/citation.cfm?id=1008841.1008855.

[2] A. Efrima and D. Peleg. Distributed models and algorithms for mobile robot systems. In *Proceedings of the 33rd conference on Current Trends in Theory and Practice of Computer Science*, SOFSEM '07, pages 70–87, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 978-3-540-69506-6. URL http://dx.doi.org/10.1007/978-3-540-69507-3_5.

[3] J. Mclurkin and J. Smith. Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots. In *in 7th International Symposium on Distributed Autonomous Robotic Systems (DARS*, 2004.

[4] A. Efrima and D. Peleg. Distributed algorithms for partitioning a swarm of autonomous mobile robots. In *Proceedings of the 14th international conference on Structural information and communication complexity*, SIROCCO'07, pages 180–194, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 978-3-540-72918-1. URL http://dl.acm.org/citation.cfm?id=1760631.1760651.

[5] L. Bayindir and E. Şahin. A Review of Studies in Swarm Robotics. *Turkish Journal of Electrical Engineering and Computer Sciences*, 15:115–147, 2007.

[6] G. Lee and N.Y. Chong. A geometric approach to deploying robot swarms. *Annals of Mathematics and Artificial Intelligence*, 52:257–280, April 2008. ISSN 1012-2443. doi: 10.1007/s10472-009-9125-x. URL http://dl.acm.org/citation.cfm?id=1527581.1527606.

[7] Y. Dieudonn, F. Petit, and V. Villain. Leader election problem versus pattern formation problem. In Nancy Lynch and Alexander Shvartsman, editors, *Distributed Computing*, volume 6343 of *Lecture Notes in Computer Science*, pages 267–281. Springer Berlin / Heidelberg, 2010. ISBN 978-3-642-15762-2. URL http://dx.doi.org/10.1007/978-3-642-15763-9_26. 10.1007/978-3-642-15763-9_26.

[8] J. Czyzowicz, L. Gasieniec, and A. Pelc. Gathering few fat mobile robots in the plane. *Theor. Comput. Sci.*, 410:481–499, February 2009. ISSN 0304-3975. doi: 10.1016/j.tcs.2008.10.005. URL http://dl.acm.org/citation.cfm?id=1497655.1498548.

[9] S. Nouyan, M. Dorigo, S. Nouyan, and M. Dorigo. March 2004chain formation in a swarm of robots, 2004.

[10] R. Gro, M. Bonani, F. Mondada, and M. Dorigo. Autonomous self-assembly in swarmbots. *IEEE Trans. Robot*, pages 1115–1130, 2006.

[11] R. Olfati-saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51:401–420, 2006.

[12] V. Trianni, S. Nolfi, and M. Dorigo. Cooperative hole avoidance in a swarm-bot, 2004.

[13] K.M. Passino. Biomimicry of bacterial foraging for distributed optimization and control. *Control Systems, IEEE*, 22(3):52 –67, jun 2002. ISSN 1066-033X. doi: 10.1109/MCS.2002.1004010.

[14] S. Nouyan, A. Campo, and M. Dorigo. Path formation in a robot swarm – self-organized strategies to find your way home, 2008.

[15] P. Muniganti and A.O. Pujol. A survey on mathematical models of swarm robotics.

[16] D. Eppstein, M.T. Goodrich, and N. Sitchinava. Foundational algorithms for computational distributed robot swarms.

[17] I. Suzuki and M. Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM Journal on Computing*, 28:1347–1363, 1999.

[18] E. Bahceci, O. Soysal, and E. Sahin. A review: Pattern formation and adaptation in multi-robot systems. *Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-03-43*, 2003.

[19] K. Sugihara and I. Suzuki. Distributed algorithms for formation of geometric patterns with many mobile robots. *Journal of robotic systems*, 13(3):127–139, 1996.

[20] T. Balch and M. Hybinette. Behavior-based coordination of large-scale robot formations. In *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*, pages 363–364. IEEE, 2000.

[21] T. Balch and M. Hybinette. Social potentials for scalable multi-robot formations. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 1, pages 73–80. IEEE, 2000.

[22] K. Fujibayashi, S. Murata, K. Sugawara, and M. Yamamura. Self-organizing formation algorithm for active elements. In *Reliable Distributed Systems, 2002. Proceedings. 21st IEEE Symposium on*, pages 416–421. IEEE, 2002.

[23] J.P. Desai. Modeling multiple teams of mobile robots: a graph theoretic approach. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 1, pages 381–386. IEEE, 2001.

[24] R. Fierro and A.K. Das. A modular architecture for formation control. In *Robot Motion and Control, 2002. RoMoCo '02. Proceedings of the Third International Workshop on*, pages 285 – 290, nov. 2002. doi: 10.1109/ROMOCO.2002.1177121.

[25] G. Lee and S. Yoon. A mobile sensor network forming concentric circles through local interaction and consensus building. *Journal of Robotics and Mechatronics*, 21:469–477, August 2009. ISSN 1883-8049. URL http://www.fujipress.jp/finder/access_check.php?pdf_filename=ROBOT002100040004.pdf&frompage=abst_page&errormode=Login&pid=2102&lang=English.

[26] P.C. Gurumohan and J. Hui. Topology design for free space optical networks. In *Computer Communications and Networks, 2003. ICCCN 2003. Proceedings. The 12th International Conference on*, pages 576 – 579, oct. 2003. doi: 10.1109/ICCCN.2003.1284227.

[27] J. Kari. Theory of cellular automata: a survey. *Theor. Comput. Sci.*, 334 (1-3):3–33, April 2005. ISSN 0304-3975. doi: 10.1016/j.tcs.2004.11.021. URL http://dx.doi.org/10.1016/j.tcs.2004.11.021.

[28] A. Ilachinski. *Cellular Automata: A Discrete Universe.* World Scientific, 2001. ISBN 9789812381835. URL http://books.google.co.uk/books?id=3Hx2lx_pEF8C.

[29] R. Hekmat. *Ad-hoc networks: fundamental properties and network topologies.* Springer, 2006. ISBN 9781402051654. URL http://books.google.co.uk/books?id=p4TwPgAACAAJ.

[30] R.P. Feynman, R.B. Leighton, and M.L. Sands. *The Feynman lectures on physics.* Number v. 1 in Addison-Wesley world student series. Addison-Wesley Pub. Co., 1963. URL http://books.google.co.uk/books?id=_ZUfAQAAMAAJ.

[31] G. Tel. *Introduction to Distributed Algorithms*. Cambridge University Press, 2000. ISBN 9780521794831. URL http://books.google.co.uk/books?id=vlpnS25qAJQC.

[32] T.S. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall communications engineering and emerging technologies series. Pearson Education, 2009. ISBN 9788131728826. URL http://books.google.co.uk/books?id=11qEWkNFFwQC.

[33] A. Rosenfeld and J.L. Pfaltz. Distance functions on digital pictures. *Pattern Recognition*, 1(1):33 – 61, 1968. ISSN 0031-3203. doi: 10.1016/0031-3203(68)90013-7. URL http://www.sciencedirect.com/science/article/pii/0031320368900137.

[34] B. Hajdu, A. Nagy and Z. Zorgo. Indexing and segmenting colour images using neighbourhood sequences. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 1, pages I – 957–60 vol.1, sept. 2003. doi: 10.1109/ICIP.2003.1247123.

[35] Sz. Farkas, J. Bajk and B. Nagy. Approximating the Euclidean circle in the square grid using neighbourhood sequences. *Pure Math. Appl. (PU.M.A.)*, 17:309–322, 2006. ISSN 1218-4586.

[36] A. Hajdu. Geometry of neighbourhood sequences. *Pattern Recognition Letters*, 24(15):2597 – 2606, 2003. ISSN 0167-8655. doi: 10.1016/S0167-8655(03)00104-1. URL http://www.sciencedirect.com/science/article/pii/S0167865503001041.

[37] A. Hajdu. Properties and applications of neighbourhood sequences. *Third Conference on Computer Graphics and Geometry*, 2005.

[38] S. Farkas, J. Bajak and B. Nagy. Approximating the Euclidean circle in the square grid using neighbourhood sequences. *ArXiv e-prints*, June 2010.

[39] A. Fazekas, A. Hajdu and L. Hajdu. Lattice of generalized neighbourhood sequences in nd. In *Image and Signal Processing and Analysis, 2003. ISPA*

2003. Proceedings of the 3rd International Symposium on, volume 1, pages 107 – 111 Vol.1, sept. 2003. doi: 10.1109/ISPA.2003.1296877.

[40] B. Nagy and R. Strand. Approximating euclidean distance using distances based on neighbourhood sequences in non-standard three-dimensional grids. In Ralf Reulke, Ulrich Eckardt, Boris Flach, Uwe Knauer, and Konrad Polthier, editors, *Combinatorial Image Analysis*, volume 4040 of *Lecture Notes in Computer Science*, pages 89–100. Springer Berlin / Heidelberg, 2006. ISBN 978-3-540-35153-5. URL http://dx.doi.org/10.1007/11774938_8. 10.1007/11774938_8.

[41] R. Strand and B. Nagy. Distances based on neighbourhood sequences in non-standard three-dimensional grids. *Discrete Applied Mathematics*, 155(4):548 – 557, 2007. ISSN 0166-218X. doi: 10.1016/j.dam.2006.09.005. URL http://www.sciencedirect.com/science/article/pii/S0166218X06003842.

[42] M. Yamashita and N. Honda. Distance functions defined by variable neighborhood sequences. *Pattern Recognition*, 17(5):509 – 513, 1984. ISSN 0031-3203. doi: 10.1016/0031-3203(84)90048-7. URL http://www.sciencedirect.com/science/article/pii/0031320384900487.

[43] M. Yamashita and T. Ibaraki. Distances defined by neighborhood sequences. *Pattern Recognition*, 19(3):237 – 246, 1986. ISSN 0031-3203. doi: 10.1016/0031-3203(86)90014-2. URL http://www.sciencedirect.com/science/article/pii/0031320386900142.

[44] B. Nagy. Metric and non-metric distances on zn by generalized neighbourhood sequences. In *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*, pages 215 – 220, sept. 2005. doi: 10.1109/ISPA.2005.195412.

[45] B. Nagy. Characterization of digital circles in triangular grid. *Pattern Recognition Letters*, 25(11):1231 – 1242, 2004. ISSN 0167-8655. doi: 10.1016/j.patrec.2004.04.001. URL http://www.sciencedirect.com/science/article/pii/S0167865504000935.

[46] B. Nagy. Finding shortest path with neighbourhood sequences in triangular grids. In *Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium on*, pages 55 –60, 2001. doi: 10.1109/ISPA.2001.938603.

[47] B. Nagy and R. Strand. Neighborhood sequences on nd hexagonal/face-centered-cubic grids. In *Proceedings of the 13th International Workshop on Combinatorial Image Analysis*, IWCIA '09, pages 96–108, Berlin, Heidelberg, 2009. Springer-Verlag. ISBN 978-3-642-10208-0. doi: 10.1007/978-3-642-10210-3_8. URL http://dx.doi.org/10.1007/978-3-642-10210-3_8.

[48] B. Nagy. Geometry of neighborhood sequences in hexagonal grid. In *Proceedings of the 13th international conference on Discrete Geometry for Computer Imagery*, DGCI'06, pages 53–64, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3-540-47651-2, 978-3-540-47651-1. doi: 10.1007/11907350_5. URL http://dx.doi.org/10.1007/11907350_5.

[49] B. Nagy and R. Strand. Approximating euclidean circles by neighbourhood sequences in a hexagonal grid. *Theor. Comput. Sci.*, 412(15):1364–1377, March 2011. ISSN 0304-3975. doi: 10.1016/j.tcs.2010.10.028. URL http://dx.doi.org/10.1016/j.tcs.2010.10.028.

[50] B. Nagy. Calculating distance with neighborhood sequences in the hexagonal grid. In Reinhard Klette and Jovia unic, editors, *Combinatorial Image Analysis*, volume 3322 of *Lecture Notes in Computer Science*, pages 98–109. Springer Berlin / Heidelberg, 2005. ISBN 978-3-540-23942-0. URL http://dx.doi.org/10.1007/978-3-540-30503-3_8. 10.1007/978-3-540-30503-3_8.

[51] R. Schneider. *Convex Bodies: The Brunn-Minkowski Theory*. Encyclopedia of Mathematics and Its Applications. Cambridge University Press, 1993. ISBN 9780521352208. URL http://books.google.co.uk/books?id=2QhT8UCKx2kC.

[52] H. Freeman. On the encoding of arbitrary geometric configurations. *Electronic Computers, IRE Transactions on*, EC-10(2):260 –268, june 1961. ISSN 0367-9950. doi: 10.1109/TEC.1961.5219197.

[53] I. Amidror. *The Theory of the Moiré Phenomenon: Aperiodic Layers*. Computational Imaging and Vision. Springer, 2007. ISBN 9781402054570. URL http://books.google.co.uk/books?id=Z_QRomE5g3QC.

[54] R. Martin, T. Nickson, and I. Potapov. Geometric computations by broadcasting automata on the integer grid. In Cristian Calude, Jarkko Kari, Ion Petre, and Grzegorz Rozenberg, editors, *Unconventional Computation*, volume 6714 of *Lecture Notes in Computer Science*, pages 138–151. Springer Berlin / Heidelberg, 2011. ISBN 978-3-642-21340-3. URL http://dx.doi.org/10.1007/978-3-642-21341-0_18. 10.1007/978-3-642-21341-0_18.

[55] R. Martin, T. Nickson, and I. Potapov. Geometric computations by broadcasting automata. *Natural Computing*, pages 1–13, 2012. ISSN 1567-7818. URL http://dx.doi.org/10.1007/s11047-012-9330-0. 10.1007/s11047-012-9330-0.

[56] T. Nickson and I. Potapov. Discrete discs and broadcasting sequences. In Jrme Durand-Lose and Nataa Jonoska, editors, *Unconventional Computation and Natural Computation*, volume 7445 of *Lecture Notes in Computer Science*, pages 235–235. Springer Berlin / Heidelberg, 2012. ISBN 978-3-642-32893-0. URL http://dx.doi.org/10.1007/978-3-642-32894-7_23. 10.1007/978-3-642-32894-7_23.

[57] J.H. Conway. *Regular Algebra and Finite Machines*. Dover Books on Mathematics Series. Dover Publications, 2012. ISBN 9780486485836. URL http://books.google.co.uk/books?id=1KAXc5TpEV8C.

[58] R. Duncan. A survey of parallel computer architectures. *Computer*, 23(2):5 –16, feb. 1990. ISSN 0018-9162. doi: 10.1109/2.44900.

[59] T. Feng. A survey of interconnection networks. *Computer*, 14(12):12 –27, dec. 1981. ISSN 0018-9162. doi: 10.1109/C-M.1981.220290.

[60] P. Linz. *An Introduction to Formal Languages and Automata*. Theory of Computation Series. Jones and Bartlett, 2001. ISBN 9780763714222. URL http://books.google.co.uk/books?id=Cgooanwdo9AC.

[61] S. Wolfram. Universality and complexity in cellular automata. *Physica D: Nonlinear Phenomena*, 10(12):1 – 35, 1984. ISSN 0167-2789. doi: 10.1016/0167-2789(84)90245-8. URL http://www.sciencedirect.com/science/article/pii/0167278984902458.

[62] M. Gerhardt, H. Schuster, and J.J. Tyson. A cellular automaton model of excitable media: Ii. curvature, dispersion, rotating waves and meandering waves. *Physica D: Nonlinear Phenomena*, 46(3):392 – 415, 1990. ISSN 0167-2789. doi: 10.1016/0167-2789(90)90101-T. URL http://www.sciencedirect.com/science/article/pii/016727899090101T.

[63] L. Hella, M. Järvisalo, A. Kuusisto, J. Laurinharju, T. Lempiäinen, K. Luosto, J. Suomela, and J. Virtema. Weak models of distributed computing, with connections to modal logic. *CoRR*, abs/1205.2051, 2012.

[64] N. J. A. Sloane. The On-Line Encyclopedia of Integer Sequences. A001481, 1995. Numbers that are the sum of 2 nonnegative squares.

[65] J. Kari. Theory of cellular automata: a survey. *Theor. Comput. Sci.*, 334: 3–33, April 2005. ISSN 0304-3975. doi: 10.1016/j.tcs.2004.11.021. URL http://dl.acm.org/citation.cfm?id=1083031.1083033.

[66] J. Mazoyer. An overview of the firing squad synchronization problem. In C. Choffrut, editor, *Automata Networks*, volume 316 of *Lecture Notes in Computer Science*, pages 82–94. Springer Berlin / Heidelberg, 1988. ISBN 978-3-540-19444-6. URL http://dx.doi.org/10.1007/3-540-19444-4_16. 10.1007/3-540-19444-4_16.

[67] J. Matoušek. *Lectures on Discrete Geometry*. Graduate Texts in Mathematics. Springer, 2002. ISBN 9780387953731. URL http://books.google.co.uk/books?id=MzFzCZAAk8MC.

[68] T. Tao and V.H. Vu. *Additive Combinatorics*. Number v. 13 in Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2006. ISBN 9780521853866. URL http://books.google.co.uk/books?id=xpimQMtn5-IC.

[69] M.M. Deza and M. Laurent. *Geometry of Cuts and Metrics*. Algorithms and Combinatorics. Springer, 1997. ISBN 9783540616115. URL http://books.google.co.uk/books?id=UQMXPi_luLIC.

[70] J.B. Yuns. Seven-state solutions to the firing squad synchronization problem. *Theoretical Computer Science*, 127(2):313 – 332, 1994. ISSN 0304-3975. doi: 10.1016/0304-3975(94)90045-0. URL http://www.sciencedirect.com/science/article/pii/0304397594900450.

[71] J. Mazoyer. A six-state minimal time solution to the firing squad synchronization problem. *Theoretical Computer Science*, 50(2):183 – 238, 1987. ISSN 0304-3975. doi: 10.1016/0304-3975(87)90124-1. URL http://www.sciencedirect.com/science/article/pii/0304397587901241.

[72] R. Balzer. An 8-state minimal time solution to the firing squad synchronization problem. *Information and Control*, 10(1):22 – 42, 1967. ISSN 0019-9958. doi: 10.1016/S0019-9958(67)90032-0. URL http://www.sciencedirect.com/science/article/pii/S0019995867900320.

[73] E.F. Moore. *Sequential machines: selected papers*. Addison-Wesley series in computer science and information processing. Addison-Wesley Pub. Co., 1964. URL http://books.google.co.uk/books?id=qt5LAAAAMAAJ.

[74] P. Bhowmick and B. B. Bhattacharya. Number-theoretic interpretation and construction of a digital circle. *Discrete Applied Mathematics*, 156(12):2381 – 2399, 2008. ISSN 0166-218X. doi: 10.1016/j.dam.2007.10.022. URL http://www.sciencedirect.com/science/article/pii/S0166218X07004817.

[75] J. Bresenham. A linear algorithm for incremental digital display of circular arcs. *Commun. ACM*, 20(2):100–106, February 1977. ISSN 0001-0782.

doi: 10.1145/359423.359432. URL http://doi.acm.org/10.1145/359423.359432.

[76] R.C. Yates. *Curves and their properties*. Classics in mathematics education. National Council of Teachers of Mathematics, 1974. URL http://books.google.co.uk/books?id=UPs-AAAAIAAJ.

[77] A. Thiaville. Extensions of the geometric solution of the two dimensional coherent magnetization rotation model. *Journal of Magnetism and Magnetic Materials*, 182(12):5 – 18, 1998. ISSN 0304-8853. doi: 10.1016/S0304-8853(97)01014-7. URL http://www.sciencedirect.com/science/article/pii/S0304885397010147.

[78] A. Hajdu and L. Hajdu. Approximating the euclidean distance using non-periodic neighbourhood sequences. *Discrete Mathematics*, 283(13):101 – 111, 2004. ISSN 0012-365X. doi: 10.1016/j.disc.2003.12.016. URL http://www.sciencedirect.com/science/article/pii/S0012365X04001116.

[79] B. Nagy. Finding shortest path with neighbourhood sequences in triangular grids. In *Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium on*, pages 55 –60, 2001. doi: 10.1109/ISPA.2001.938603.

[80] B. Nagy. Geometry of neighborhood sequences in hexagonal grid. In Attila Kuba, Lszl Nyl, and Klmn Palgyi, editors, *Discrete Geometry for Computer Imagery*, volume 4245 of *Lecture Notes in Computer Science*, pages 53–64. Springer Berlin / Heidelberg, 2006. ISBN 978-3-540-47651-1. URL http://dx.doi.org/10.1007/11907350_5. 10.1007/11907350_5.