

UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA MATEMÁTICA

**INFLUENCIA DE LA RED ESPACIAL EN EL COMPORTAMIENTO DE
UN SISTEMA DINÁMICO: LA HORMIGA DE LANGTON**

ANAHÍ GAJARDO SCHULZ

2001

UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA MATEMÁTICA

**INFLUENCIA DE LA RED ESPACIAL EN EL COMPORTAMIENTO DE
UN SISTEMA DINÁMICO: LA HORMIGA DE LANGTON**

ANAHÍ GAJARDO SCHULZ

Profesores Guías	:	Sr. Eric Goles C. Sr. Jacques Mazoyer
Profesores Intengrantes	:	Sra. Marie-Pierre Beal Sr. Leonid Bunimovich Sr. Servet Martinez Sr. Michel Morvan
Profesor Invitado	:	Sr. Leopoldo Bertossi

TESIS PARA OPTAR AL GRADO DE
DOCTOR EN CS. DE LA ING., MENC. MODELACION MATEMATICA

SANTIAGO-CHILE
Junio, 2001

RESUMEN DE LA TESIS
PARA OPTAR AL GRADO DE
DOCTOR EN CS. DE LA INGENIERÍA.
MENC. MODELACIÓN MATEMÁTICA
POR : ANAHÍ GAJARDO SCHULZ
FECHA : 1 DE JUNIO DE 2001
PROF. GUÍAS : SR. ERIC GOLES y
SR. JACQUES MAZOYER

INFLUENCIA DE LA RED ESPACIAL EN EL COMPORTAMIENTO DE UN SISTEMA DINÁMICO: LA HORMIGA DE LANGTON

En la presente tesis se estudia un sistema dinámico particular, la hormiga de Langton, que tiene interés en el ámbito de física estadística. El sistema consiste en un grafo en que cada vértice lleva asociado un estado binario (blanco o negro), y en un agente (la hormiga) que se desplaza por ellos y los cambia de estado. El movimiento de la hormiga está completamente determinado por el estado de los vértices que visita. Este sistema ha sido estudiado desde hace más de una década. Fenómenos curiosos e interesantes han sido descubiertos, pocos de los cuales tienen una explicación teórica.

El comportamiento del sistema depende fuertemente del grafo. El objetivo principal de este trabajo de tesis es conocer esta dependencia, desde un punto de vista teórico, para ciertos grafos.

A través de métodos combinatoriales se demostró que la trayectoria de la hormiga es simétrica bajo reflexión cuando en un comienzo, en la grilla hexagonal, todos los vértices están en el mismo estado. Además, en el caso de los grafos $\Gamma(k, d)$ se demostró que si tienen grado mayor o igual que 5, la hormiga no puede alcanzar todos los vértices, y que si en un comienzo sólo un número finito de vértices está en estado negro, entonces a partir de un momento la trayectoria de la hormiga es “regular”.

Se demostró que el sistema es universal cuando el grafo es la grilla hexagonal o cuadrada. La demostración de esto se llevó a cabo mediante la simulación de puertas y circuitos lógicos.

Finalmente, el sistema se estudió desde el punto de vista de los sistemas dinámicos simbólicos, clasificando los grafos $\Gamma(k, d)$ según diferentes propiedades del sistema, tales como la transitividad, soficidad, etc. .

Los resultados encontrados en los distintos contextos considerados coinciden en la existencia de una clara diferencia entre los grafos $\Gamma(k, d)$ de grado 3 o 4 y los de grado mayor o igual que 5. Mientras en los primeros el sistema presenta una alta complejidad, en los segundos su comportamiento es más simple y predecible.

Agradecimientos

Al profesor Jacques Mazoyer, por su dedicado apoyo científico y personal.

Al profesor Eric Goles por haberme motivado a seguir este camino.

Al grupo MC2 de el Laboratoire de l'Informatique du Parallélisme de la École Normale Supérieure de Lyon donde realicé parte de esta tesis (en cotutela), por la solidaridad y presteza con que sus integrantes me dieron ayuda cada vez que la requerí.

Al los profesores y secretarias del Departamento de Ingeniería Matemática, por su entrega a la labor docente de la cual he sido beneficiaria. Agradezco especialmente la amistad que algunos de ellos me han brindado.

A mis amigos y familiares por su generosidad y lealtad.

Al programa de becas de doctorado de CONICYT, al programa de cooperación franco chileno ECOS, y al Ministère de l'Éducation Nationale de la Recherche et de la Technologie francés, por el apoyo Económico que hizo posible el desarrollo de esta tesis.

A todos ellos estaré por siempre profundamente agradecida.

Contents

1	Introducción	11
1.1	La hormiga de Langton	11
1.2	Generalización la regla de transición	13
1.3	Comprender la hormiga de Langton	17
1.4	Resultados combinatoriales	19
1.4.1	Noción de esquina	19
1.4.2	“Baldosas de Truchet”	20
1.4.3	Caso de los grafos $\Gamma(k, d)$ con $d \geq 5$	23
1.4.4	Resumen de propiedades en los grafos $\Gamma(k, d)$	25
1.4.5	Generalización de la hormiga en la línea	25
1.5	La hormiga como un sistema simbólico	26
1.6	La hormiga como un sistema de cálculo	28
1.7	Conclusiones	30
1.8	Publicaciones	32
2	Definitions and basic notions	33

2.1	The ant's rule	33
2.2	Graphs	35
3	Some particular combinatorial properties	37
3.1	Graphs of even degree	37
3.2	Graphs of degree 3	41
4	$\Gamma(k, d)$ graphs, with $d \geq 5$	51
4.1	Main result	52
4.2	Eventually regular trajectories	68
5	Symbolic Dynamics	71
5.1	Definitions on symbolic dynamics	73
5.2	Ant's properties	75
5.3	Towards universality	84
5.4	Discussion	87
6	Universality	89
6.1	Universality in $\Gamma(k, d)$ graphs with $d = 3$ and $d = 4$	90
6.1.1	Construction of circuits	90
6.1.2	Embedding in infinite regular graphs	92
6.1.3	Computational Complexity	95
6.1.4	Conclusion	96
6.2	Decidability	98

7	The line	101
7.1	Class B	104
7.1.1	Class B1	105
7.1.2	Class B2	107
7.2	Class U	109
7.2.1	The class U1	109
7.2.2	The class U2	116
7.3	Conclusions	118
7.4	Simulations	119

Chapter 1

Introducción

1.1 La hormiga de Langton

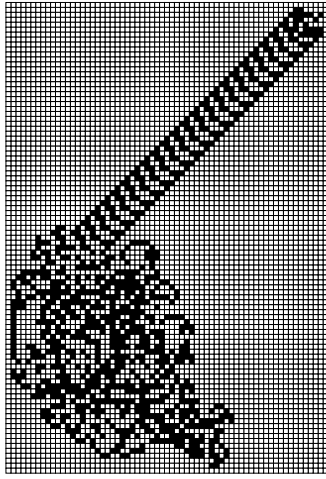
En las últimas décadas, se han introducido diversos sistemas dinámicos discretos simples (pilas de arena, autómatas celulares, etc.). Ellos son de interés en informática, ya que, por su simplicidad, se prestan al análisis teórico, y al mismo tiempo su comportamiento global es complejo y tiene interpretaciones desde otras ciencias como física y biología.

La hormiga de Langton aparece por primera vez en la década del 80 [29] como un modelo de vida artificial. Su definición informal es muy natural:

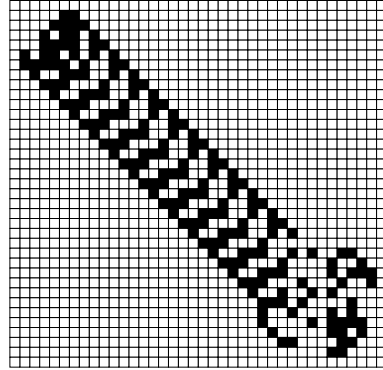
Una hormiga se desplaza por los arcos de un grafo planar cuyos vértices (células) están coloreados blancos o negros. Al llegar a cada vértice, ella dobla lo más posible a la derecha (respectivamente a la izquierda), si el vértice es de color negro (respectivamente blanco). Además, al pasar por cada vértice, hace que éste cambie de color.

Fue definida originalmente en la grilla cuadrada infinita. Su comportamiento en ese caso se muestra complicado y aún sin explicación: a partir de una configuración donde todas las células están del mismo color, la trayectoria de la hormiga presenta simetría rotacional durante las primeras 500 iteraciones; luego su comportamiento se muestra

completamente errático hasta la iteración 10 150, a partir de la cual la trayectoria se vuelve regular, desplazándose constantemente en una de las cuatro diagonales, e imprimiendo un patrón periódico que llamamos escalera (ver Figura 1.1(a)). Además, en todas las simulaciones, se observa que cada vez que la hormiga parte de un motivo finito, surge este mismo movimiento periódico (ver la Figura 1.1(b))¹ como ejemplo). Sin embargo, este hecho experimental, aún no ha sido demostrado. Es debido a esta suerte de comportamiento emergente, producto de la interacción de la hormiga con su propia traza, que el sistema fue llamado "hormiga".



(a)



(b)

Figure 1.1: (a) Configuración en la iteración 12 321, cuando parte con todas las células del mismo color. (b) Configuración en la iteración 1 527, a partir de una configuración con 2 células en color negro.

Este fenómeno nos hace recordar otros sistemas dinámicos discretos que presentan comportamientos similares, cuya demostración es igualmente un problema abierto. Tal es el caso de las series de Collatz [11] y de las "lombrices programadas" [14, 15].

En física, la hormiga de Langton ha sido estudiada de manera independiente como un paradigma en propagación de señales en medios aleatorios. Específicamente, como un

¹Estas figuras fueron obtenidas usando el software 'ant.c', creado por J. Propp [40].

modelo de gas de Lorentz en una grilla bidimensional, en el que una partícula interactúa con obstáculos que llenan el espacio y afectan su trayectoria [10]. El interés, en este caso, es conocer el comportamiento medio de la hormiga (partícula) en el largo plazo, cuando la configuración inicial es aleatoria.

Buscando comprender este fenómeno se han realizado diversos estudios -tanto experimentales como analíticos-, que han llevado a la introducción de nuevas herramientas de análisis, tales como las "baldosas de Truchet" y el "grafo de diagonales". El primer y más importante resultado teórico, obtenido por L. A. Bunimovich y S. Troubetzkoy [4], demuestra que, en la grilla cuadrada, la trayectoria de la hormiga no es acotada, independientemente de la configuración inicial.

En [6, 7, 46] y, en general, en los trabajos en física, se consideran distintos tipos de movimiento de la hormiga y distintas grillas; en todos los casos se producen diferentes comportamientos. En [20, 3], se consideran varios estados para cada célula, dando lugar a un número infinito de reglas, y se han hecho pruebas con una veintena de ellas. Aparecen escaleras (como en el caso original), trayectorias simétricas bajo reflexión (ver Figura 1.2), trayectorias en espiral (ver Figura 1.3), así como trayectorias sin ningún patrón de crecimiento aparente. Uno de los resultados más importantes en esta dirección [12] demuestra que para ciertas reglas la trayectoria de la hormiga (en una configuración uniforme) presenta simetría bajo reflexión [21, 22].

La dinámica de la hormiga está fuertemente relacionada con la topología del grafo subyacente. En la grilla hexagonal, se aprecian diversos comportamientos, desde trayectorias acotadas, hasta trayectorias aparentemente erráticas, pasando por trayectorias simétricas bajo reflexión [46], sin embargo, en ninguna simulación se forma una escalera como en el caso de la grilla cuadrada. En la grilla triangular, la trayectoria de la hormiga está restringida a una banda de ancho dos; allí, la hormiga se desplaza avanzando a una velocidad mínima, de una célula por cada siete unidades de tiempo, en una dirección fija [25] (ver una simulación en la Figura 1.4).

1.2 Generalización la regla de transición

Cuando buscamos en generalizar la regla de transición de la hormiga a un grafo diferente de la grilla cuadrada, tenemos que determinar cuáles son las características esenciales que nos interesa preservar. Pensamos que la simplicidad de la hormiga radica, principalmente, en el hecho que, en si misma, no posee memoria ni conocimiento alguno del grafo sobre el cual se desplaza. La hormiga no sabe en qué coordenada se encuentra ni

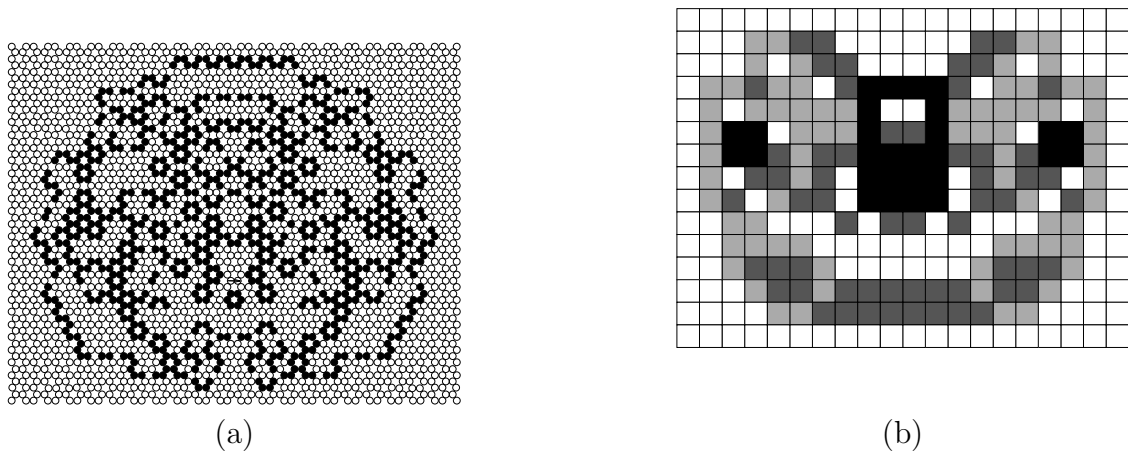


Figure 1.2: Trayectorias simétricas. (a) Iteración 31 260 partiendo con todas las células en estado blanco en la grilla hexagonal. (b) Iteración 3 972 de una generalización de la regla con 4 estados en la grilla cuadrada, partiendo con todas las células en estado blanco.

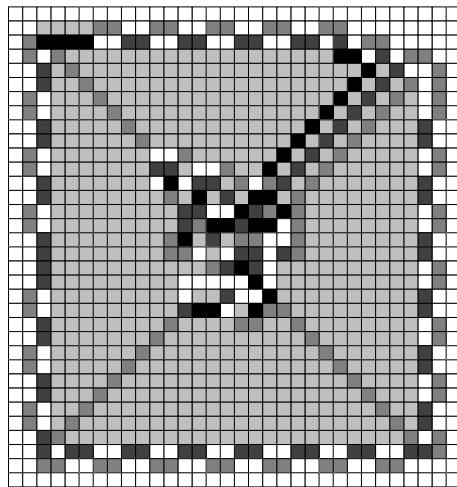


Figure 1.3: Iteración 5 479 de una generalización de la regla con 5 estados, partiendo de la grilla cuadrada blanca.

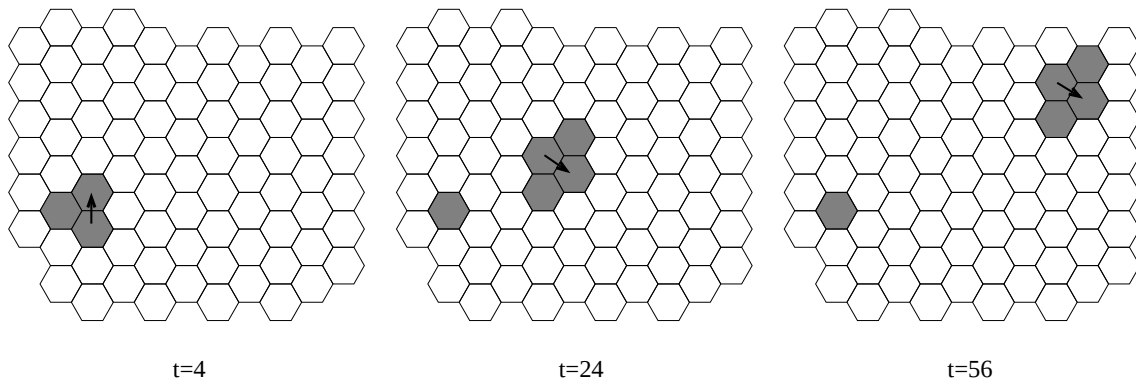


Figure 1.4: Una simulación en la grilla triangular partiendo en el tiempo 0 con todas las células en color blanco.

dónde están el "sur" y el "norte", lo único que ella puede distinguir es cuál es su derecha y su izquierda, y la única información que utiliza, para decidir su próximo movimiento, es el estado de la célula a la que está apuntando.

Otra característica que nos interesa preservar es la minimalidad del número de estados de las células: dos colores. Finalmente, al definir la regla de la hormiga, nos gustaría que ésta fuese la misma para cualquier grafo, dicho de otra manera, que la hormiga no necesite conocer ninguna característica del grafo para decidir su movimiento.

En la definición informal dada en la sección anterior, el movimiento de la hormiga está determinado por una noción geométrica: "ella dobla lo más posible a la derecha (respectivamente a la izquierda) si el vértice está de color negro (respectivamente blanco)". Esta regla puede generalizarse directamente a cualquier grafo, siempre que hablar de derecha e izquierda tenga sentido. Formalmente, la noción de derecha e izquierda se define cuando se está *sobre* una superficie de dimensión 2, que no necesita ser orientable, basta que sea localmente orientable. Entonces, la regla puede ser definida para cualquier grafo inyectable en una superficie como ésta (ver Figura 1.5). Basta especificar, para cada célula, su orientación, la cual es fija, y agregar una variable de estado más a la hormiga, indicando de que lado de la superficie se encuentra, con respecto a la orientación de la célula a la que apunta. Sin embargo, cuando la superficie es orientable, la hormiga permanece siempre del mismo lado de la superficie; entonces, basta especificar este dato al comienzo y los sentidos derecho e izquierdo quedan definidos, para todas las células del grafo, de manera única.

En este trabajo, restringiremos la generalización a grafos planares, es decir, que pueden

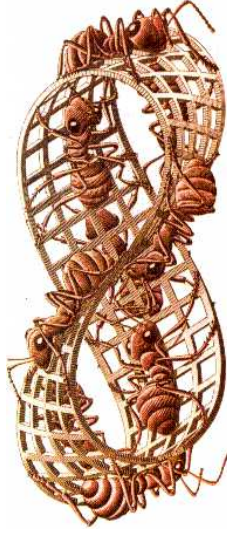


Figure 1.5: La hormiga en la cinta de Moebius (M.C. Escher).

ser inyectados en el plano. Nos restringimos, también, a grafos no dirigidos y simples ². La definición formal que adoptamos es la siguiente:

Definición. *Dada la representación planar de un grafo (V, E) . Si en una iteración dada, la hormiga está en la posición $h_t = (i, j)$, sobre el arco $\{i, j\}$, apuntando a la célula j , y si los estados de cada célula en V están dados por la configuración $c_t : V \rightarrow \{\text{blanco}, \text{negro}\}$, en la iteración siguiente, su posición y el estado de las células estarán dadas por:*

$$h_{t+1} = \begin{cases} (j, l) & \text{si } c_t(j) = \text{blanco} \\ (j, r) & \text{if } c_t(j) = \text{negro} \end{cases}$$

$$c_{t+1}(j) = \begin{cases} \text{negro} & \text{if } c_t(j) = \text{blanco} \\ \text{blanco} & \text{if } c_t(j) = \text{negro} \end{cases}$$

donde $\{j, l\}$ es el arco que está a la izquierda de $\{i, j\}$, con respecto a j y $\{j, r\}$ es el que está a la derecha.

²En la definición de la regla que estamos considerando, exigir que el grafo sea simple no es una restricción demasiado fuerte, ya que la dinámica en un grafo no simple puede ser simulada en un grafo simple, como lo demostramos en [19]

Notemos que el movimiento de la hormiga está ligado a la representación del grafo en el plano; a priori, la dinámica del sistema puede ser diferente en distintas representaciones de un mismo grafo. Esto nos permite usar no sólo herramientas combinatoriales, sino también geométricas.

El grafo subyacente, (V, E) , puede tener un número infinito de arcos y células. La única restricción aparentemente necesaria para que el movimiento esté bien definido, es que cada célula tenga un número finito de arcos adyacentes. También podemos suponer que (V, E) es conexo, pues la hormiga no puede pasar de una componente conexa a otra. Estas dos condiciones implican que V es enumerable. Finalmente, excluirémos de nuestro estudio los grafos finitos, ya que en ellos la trayectoria de la hormiga es siempre periódica³.

Resumiendo, la hormiga se desplaza sobre la representación de un grafo planar, finitamente conectado, enumerable y conexo.

1.3 Comprender la hormiga de Langton

El fenómeno que atrajo nuestra atención, fue la fuerte dependencia entre la dinámica de la hormiga y la topología de la grilla; tanto su complejidad, como el tipo de trayectorias varían enormemente de una grilla a otra. De alguna manera, el comportamiento de la hormiga entrega información sobre el grafo en que se mueve. Como Langton cita (antes de que la regla fuese generalizada a otras grillas): la trayectoria de la hormiga es un reflejo del medio en que se desplaza.

Estudiar la hormiga en diferentes grafos permite comprenderla mejor, y encontrar las características que van mas allá del grafo subyacente. Al mismo tiempo, este estudio pone en evidencia algunas propiedades de estos grafos.

En lugar de estudiar la hormiga en grafos planares arbitrarios, preferimos hacerlo en una pequeña familia de grafos que poseen una alta regularidad: **los grafos** $\Gamma(k, d)$. La representación de un grafo planar, divide el plano en sectores delimitados por sus arcos. Estos sectores se llaman *caras*. Considerando esto, se define un grafo $\Gamma(k, d)$ como un grafo planar, donde cada célula tiene grado d , y cada cara tiene cardinalidad k , con la condición adicional sobre su representación en el plano, que cada bola de radio finito contenga sólo un número finito de células. Para cada par (k, d) , existe un único grafo $\Gamma(k, d)$. Si $k \leq 5$ y $d \leq 3$ o $k \leq 3$ y $d \leq 5$, $\Gamma(k, d)$ es un grafo finito. Si

³Un estudio del orden de magnitud de este periodo se encuentra en [19]

$(k, d) = (3, 6)$, $(4, 4)$ o $(6, 3)$, entonces $\Gamma(k, d)$ se puede representar en el plano euclidiano de manera que todas las caras tengan el mismo tamaño; su representación, en estos casos, corresponde a las grillas triangular, cuadrada y hexagonal, respectivamente. Los casos en que la representación de $\Gamma(k, d)$ no puede satisfacer este criterio, se denominan grafos *hiperbólicos*, y corresponden a embaledos del plano hiperbólico con k -gonos. La Figura 1.6 muestra un ejemplo de un grafo $\Gamma(k, d)$ hiperbólico.

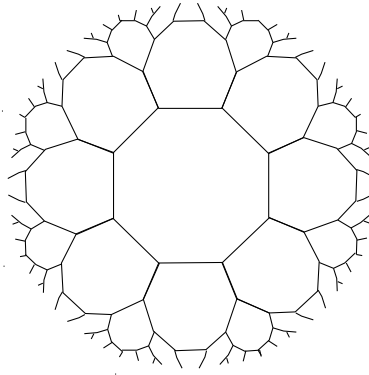


Figure 1.6: El grafo $\Gamma(8, 3)$

Tenemos así, una generalización de las grillas regulares (en las que todas las caras son idénticas, salvo por rotaciones) a otras grillas, correspondientes al plano hiperbólico (donde todas las caras son idénticas, salvo por semejanza). A todo par (k, d) corresponden muchas representaciones $\Gamma(k, d)$; sin embargo, ellas son implícitamente iguales (ya que tienen las mismas propiedades geométricas, es decir, la misma orientación). El hecho que la representación considerada (que por abuso de notación la llamamos grafo $\Gamma(k, d)$) esté bien definida, es fundamental para poder enunciar correctamente la regla de la hormiga.

Intuitivamente, estos grafos $\Gamma(k, d)$ son más complicados que las grillas regulares: el aumento de tamaño de las bolas es cuadrático para los grafos $\Gamma(3, 6)$, $\Gamma(4, 4)$ o $\Gamma(6, 3)$, y se hacen exponenciales para los demás $\Gamma(k, d)$ no finitos. Por otra parte, estos $\Gamma(k, d)$, son grafos sobre los que se han escrito diferentes trabajos, y cuyas características combinatoriales empiezan a ser conocidas. Finalmente, la regularidad de estos grafos nos permite poner en evidencia la relación entre la dinámica de la hormiga y dos parámetros importantes del grafo, a saber, su grado y el tamaño de sus caras.

Encaminamos nuestro estudio desde tres frentes diferentes:

- Adaptar las herramientas combinatoriales existentes para obtener resultados generales, poniendo en evidencia propiedades particulares de los grafos.
- Expresar, en el lenguaje de sistemas dinámicos, las propiedades ligadas a los diferentes valores de k y d .
- Estudiar el poder de cálculo del sistema, desde el punto de vista de la calculabilidad y de la complejidad.

1.4 Resultados combinatoriales

Como los grafos son objetos discretos, el análisis combinatorial aparece como el método más natural de abordar la problemática de la hormiga. Se hizo un análisis microscópico de la dinámica de la hormiga, específicamente de su dependencia en el grado (d) del grafo. El estudio realizado permitió mostrar propiedades de la hormiga que dependen de ciertas características específicas del grafo. Nuestros resultados no sólo se aplican a grafos bi-regulares sino también a (representaciones de) grafos generales, satisfaciendo algunas restricciones de regularidad. En nuestro estudio nos basamos en resultados clásicos de teoría de grafos, así como en resultados más nuevos sobre grafos hiperbólicos.

1.4.1 Noción de esquina

En trabajos anteriores sobre la grilla cuadrada, se observó un hecho fundamental:

Al colorear las caras como un tablero de ajedrez, nos damos cuenta que al mirar las dos caras adyacentes a la hormiga, el color de la cara de la izquierda es siempre el mismo y el de la derecha también. Es decir, si al comienzo la hormiga tiene a la derecha una cara blanca y a la izquierda una cara negra, esta situación se mantendrá a través de toda la evolución del sistema, independientemente de la distribución de estados de las células.

Esto implica que:

Dada una posición inicial de la hormiga, es posible determinar, para cada arco, el sentido en el cual la hormiga lo visitará; y este sentido será siempre el mismo.

En la Sección 3.1, observamos que esto se generaliza a cualquier grafo de grado par, y más aún, de acuerdo a los resultados del Capítulo 4, esto es cierto también para todos los grafos $\Gamma(k, d)$ con $d \geq 5$. En función de la noción de esquina, introducida en [5], podemos expresar diversas características de la trayectoria de la hormiga.

Dado un grafo planar G , y una de sus representaciones en el plano, y dado un subgrafo S , entonces, una esquina de S es una célula u , tal que su grado en G es estrictamente mayor que su grado en S , y que este último es 0, 1 o 2, y que si es 2, los arcos correspondientes están el uno al lado del otro (ver Figura 1.7).

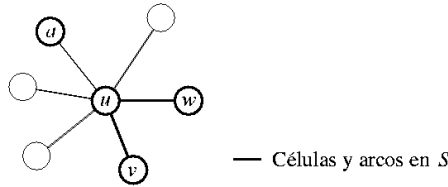


Figure 1.7: u es una esquina de S

Se demuestra que:

- Las esquinas del conjunto de células que la hormiga visita (S), corresponden a las células que son visitadas una sola vez.
- En los grafos $\Gamma(k, d)$ con $d \geq 4$, el conjunto de células que son visitadas un número infinito de veces (S'), no tiene esquinas.
- Si en un grafo cualquiera, la trayectoria de la hormiga es acotada (periódica), entonces los arcos adyacentes a las esquinas de la configuración son recorridos en ambos sentidos.

Del último punto, se concluye que en los grafos $\Gamma(k, d)$ con $d \geq 4$, la trayectoria de la hormiga nunca es acotada, independientemente de la configuración inicial.

1.4.2 “Baldosas de Truchet”

El concepto “baldosa de Truchet” fue introducido por C. Pickover [39] y utilizado por B. Rümmler [21] para demostrar la aparición de trayectorias simétricas en una de las generalizaciones de la hormiga en la grilla cuadrada.

Consiste en reemplazar las células por fichas que indican si la célula está en estado blanco o negro. Estas fichas reciben el nombre de baldosas de Truchet y se muestran en la Figura 1.8. Notemos que la baldosa de Truchet, que corresponde a una célula en estado blanco, es una reflexión especular de la baldosa de Truchet que corresponde a una célula en estado negro.

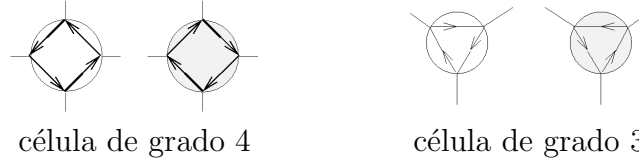


Figure 1.8: Baldosas de Truchet para células de grado 3 y 4.

El interés de este embaldosado es que permite ver a las células como el principio del movimiento del sistema, interpretando a la hormiga como un objeto sometido a la acción de éstas.

La regla dice que cuando la hormiga apunta a una célula, en la iteración siguiente apuntará a una de sus vecinas, la cual depende del estado de la primera. Desde este punto de vista, la célula puede ser vista como un agente que toma la hormiga y la mueve a su nueva posición. Gráficamente, las flechas al interior de una baldosa de Truchet se interpretan como rieles que guían a la hormiga al siguiente arco correspondiente. En el embaldosado, la hormiga aparece en la frontera de dos baldosas de Truchet, justo al comienzo de una de las flechas. Así, una configuración de estados en un grafo, se reemplaza por un embaldosado del plano con baldosas de Truchet, como lo muestra la Figura 1.9.

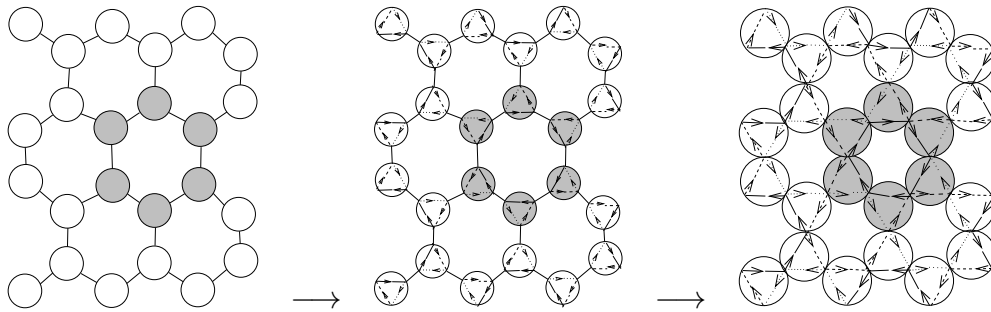


Figure 1.9: Cómo las células son reemplazadas por baldosas de Truchet.

El interés principal de esta representación reside en la noción de *contorno*. Examinemos las Figuras 1.9 y 1.10; aparecen líneas de flechas, separadas por los bordes de las células. Hemos dibujado algunas flechas con líneas punteadas y otras con líneas segmentadas para destacar esta observación. Supongamos que coloreamos las flechas, de manera que dos flechas contiguas tengan el mismo color; las series de flechas de un mismo color forman lo que llamaremos *contornos*. Reservaremos el nombre de *contorno principal* para el contorno donde está la hormiga. La Figura 1.10 muestra dos embaldosados más en las grillas triangular y hexagonal; marcamos las células en estado negro, ensombreciendo las baldosas de Truchet.

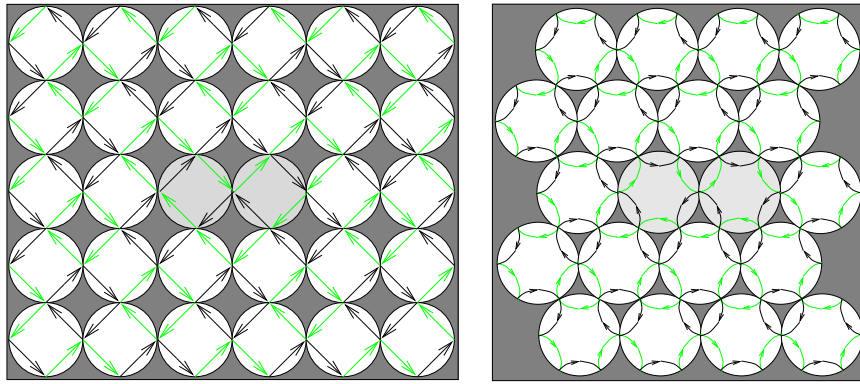


Figure 1.10: Embaldosado con baldosas de Truchet de las grillas cuadrada y triangular. En este caso, dos de las células están en estado negro.

Así como una flecha indica el movimiento de la hormiga en una iteración, un contorno indica el movimiento de la hormiga en varias iteraciones. Cuando la hormiga pasa sobre una célula, la cambia de estado. En el embaldosado, esto se traduce en que la hormiga “da vuelta” las baldosas de Truchet al pasar sobre ellas. Dada una configuración y el contorno principal, podemos prever que la hormiga seguirá el contorno, hasta que una de las células visitadas se repita.

Los contornos permiten prever la trayectoria de la hormiga en varios pasos, sólo con mirar la configuración presente. También nos permiten realizar lo contrario: podemos dibujar la trayectoria que deseamos usando baldosas de Truchet, y luego, reemplazando estas por las células en los estados correspondientes, obtenemos la configuración que producirá la trayectoria deseada.

En términos de predecir la trayectoria de la hormiga, los contornos que pasan sólo una vez por cada célula, nos dan más información que aquellos que pasan más de una vez.

Sabemos que la hormiga seguirá el contorno entero. Llamaremos a estos contornos *simples*.

La Sección 3.2 habla sobre los grafos regulares de grado 3. El resultado allí demostrado dice que, si los contornos son todos simples y los podemos colorear usando sólo tres colores -de manera que nunca se topen dos contornos del mismo color-, entonces la hormiga recorre completamente el contorno principal, y si vuelve al punto de partida, los contornos han cambiado, pero las dos propiedades que tenían al comienzo se conservan.

Esto implica que para una gran cantidad de configuraciones, entre las cuales se cuenta la configuración donde todas las células tienen el mismo estado, la hormiga vuelve una infinidad de veces al origen y explica la aparición reiterativa de configuraciones simétricas.

1.4.3 Caso de los grafos $\Gamma(k, d)$ con $d \geq 5$

El Capítulo 4 contiene el más relevante de los resultados combinatoriales, el cual se refiere a grafos $\Gamma(k, d)$ infinitos de grado mayor que 5, lo que se expresa por: $d = 5$ y $k \geq 4$ o bien $d \geq 6$ y $k \geq 3$.

El resultado es:

Dado un grafo $\Gamma(k, d)$ que satisface que: $d = 5$ y $k \geq 4$ o bien $d \geq 6$ y $k \geq 3$. Dada una configuración inicial tal que las dos células, entre las cuales se encuentra la hormiga, están en el mismo estado. Entonces la trayectoria de la hormiga está restringida a un subgrafo como el de la Figura 1.11

El resultado se basa en 3 proposiciones que describimos a continuación:

La primera establece que los únicos ciclos simples que la hormiga puede hacer tienen largo k , y corresponden a rodear una cara. La demostración se basa en un análisis del largo de los ciclos que puede hacer una hormiga y de la cantidad de células que estos encierran. Estos dos factores están relacionados por la fórmula de Euler. La condición de que el ciclo debe ser posible de recorrer por una hormiga, establece una segunda relación entre estos dos factores, la cual contradice la fórmula de Euler cuando $d = 5$ y $k \geq 4$ o bien $d \geq 6$ y $k \geq 3$.

La segunda proposición muestra que las trayectorias simples que puede hacer una hormiga, desde una posición inicial fija, no cubren todo el grafo, sino que, más precisamente, con-

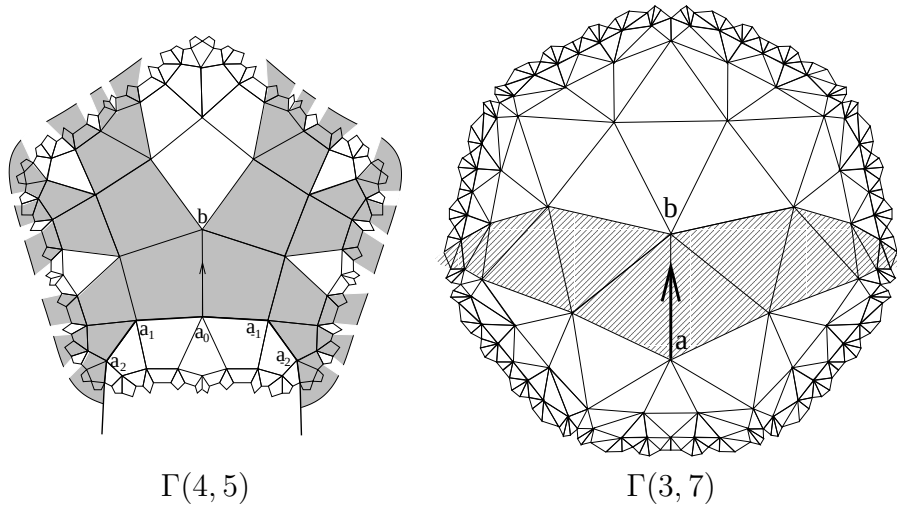


Figure 1.11: Partiendo del arco (a, b) , la hormiga sólo puede alcanzar las células en la zona sombreada.

forman un árbol como el que se muestra en la Figura 1.11 para los grafos $\Gamma(4, 5)$ y $\Gamma(3, 7)$.

La dificultad consiste en generalizar el resultado precedente a cualquier trayectoria de la hormiga. Esto se logra con la tercera proposición; la idea de la demostración, es mostrar que para cada configuración inicial c y para cada número n , existe otra configuración inicial que produce una trayectoria simple de la hormiga durante m pasos, y que ambas configuraciones llevan a la hormiga a la misma posición. Esta demostración es delicada y se basa en varias recurrencias encadenadas.

Un hecho significativo es que el grado de el árbol no depende del grado del grafo, sino que solamente de la cardinalidad de sus caras. Por esto, los resultados de la Sección 3.1 son válidos en todos los grafos infinitos de grado mayor o igual que 5.

Esta fuerte restricción, simplifica enormemente el análisis de la trayectoria de la hormiga y permite demostrar que ésta alcanza siempre un comportamiento regular, cuando parte de una configuración inicial finita.

1.4.4 Resumen de propiedades en los grafos $\Gamma(k, d)$

La siguiente tabla resume las propiedades combinatoriales obtenidas en los grafos $\Gamma(k, d)$ infinitos.

	$d = 3$	$d = 4$	$d \geq 5$
Si el $\Gamma(k, d)$ es infinito	Trayectorias acotadas y no acotadas	Trayectorias no acotadas y la hormiga pasa en un solo sentido por cada arco	
	la hormiga puede ir a cualquier célula del grafo		la hormiga está restringida a un árbol
Si el $\Gamma(k, d)$ es euclidiano y se parte de una configuración homogénea	Simetría bajo reflexión	escalera	escalera sin rastro (deslizador)

1.4.5 Generalización de la hormiga en la línea

La generalización de la regla de la hormiga, adoptada en los primeros capítulos, tiene como resultado una dinámica extremadamente simple en el caso de la línea; la hormiga se desplaza simplemente en una dirección fija. Por lo tanto, decidimos considerar otras generalizaciones, tratando de no apartarnos del estilo de la regla original. La dinámica de las reglas obtenidas resultó ser de fácil análisis. Por lo que nos decidimos a agregar un grado de libertad más: la posibilidad que uno de los dos estados de las células sea fijo y no cambie con la presencia de la hormiga.

Se analizaron 24 reglas diferentes, logrando especificar el comportamiento de la hormiga en cada una de ellas, con una alta precisión. Las 24 reglas fueron clasificadas en las siguientes 4 clases, las que consideran su comportamiento sobre configuraciones en que todas las células, salvo un número finito de ellas, están en el mismo estado:

- *Reglas que admiten trayectorias acotadas*
 - *Reglas que admiten sólo periodos de largo 2 o 4.* Cuando la hormiga parte sobre un patrón específico, se desplaza constantemente en una dirección,

y cuando este patrón se termina, la hormiga cae en un comportamiento periódico.

- *Reglas que admiten periodos de largo arbitrario.* Cuando la hormiga parte sobre un patrón específico, se desplaza constantemente en una dirección, dejando el patrón intacto. Cuando éste patrón se termina, la hormiga rebota y continúa en la dirección opuesta.
- *Reglas que no admiten trayectorias acotadas.*
 - *A partir de configuraciones finitas, la hormiga recorre todas las células del espacio.* Cuando la hormiga parte sobre un patrón alternado blanco y negro, se desplaza constantemente en una dirección fija, invirtiendo el patrón a su paso; al encontrar una perturbación, rebota y extiende el patrón en una célula.(ver la Figura 1.12).
 - *Existen configuraciones iniciales finitas, tal que la hormiga no recorre todas las células del espacio.* Cuando la hormiga llega a una zona en que todas las células están en el mismo estado, puede adquirir un comportamiento regular avanzando en una dirección fija.

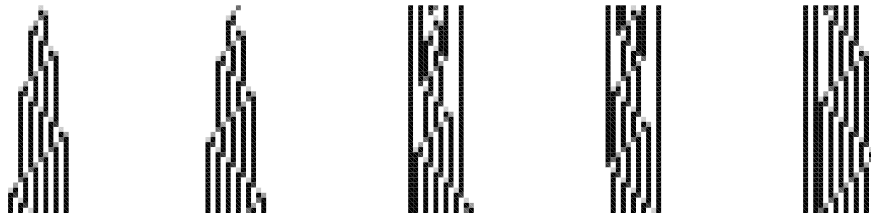


Figure 1.12: Hormiga recorriendo el interior de una parábola.

1.5 La hormiga como un sistema simbólico

Nuestro trabajo en este ámbito consistió en

- 1) encontrar una representación adecuada,
- 2) evaluar sus propiedades dinámicas.

Para esto, nos basamos en resultados previos y en los resultados obtenidos a través de los métodos combinatoriales.

La hormiga de Langton puede ser formalizada de diversas maneras. Dependiendo de cómo sea representado el estado del sistema, sus propiedades dinámicas serán diferentes. Así, en algunos modelos el sistema es transitivo y en otros no; en algunas, el espacio de fases es compacto y en otras no [5, 28].

Basamos nuestra modelación en el siguiente hecho: la diferencia entre una configuración y la siguiente es muy pequeña y puede ser codificada en un solo dígito; sólo cambia la posición de la hormiga y el estado de una célula, y estos dos cambios están directamente correlacionados. Entonces, podemos codificar toda la evolución del sistema, guardando solamente esta información en cada iteración.

La idea consiste en ir guardando la dirección en que la hormiga dobla en cada iteración; esto define una secuencia $(x_i)_{i \in \mathbb{N}}$ donde x_i es la dirección en que la hormiga dobla en la iteración i . Así, para cada configuración inicial, queda definida una secuencia infinita de caracteres R y L (“a la derecha” y “a la izquierda”) correspondiente a los giros que hará la hormiga a través del tiempo. Recíprocamente, una secuencia de giros, junto con una posición inicial, describen completamente la trayectoria de la hormiga y el estado de las células que son visitadas. Las secuencias $(x_i)_{i \in \mathbb{N}}$ contienen toda la información pertinente sobre la dinámica de la hormiga.

Formalmente, dado un grafo G , consideramos el conjunto de secuencias infinitas definidas en el parrafo anterior:

$$H_G = \left\{ (x_i)_{i \in \mathbb{N}} \in \{L, R\}^{\mathbb{N}} \mid \text{la hormiga dobla hacia } x_i \text{ en la iteración } i \right\}$$

Considerando H_G como espacio de fases, provisto de la topología producto usual y la función de traslación $\sigma((x_i)_{i \in \mathbb{N}}) = (x_{i+1})_{i \in \mathbb{N}}$ como función de transición, se obtiene un sistema simbólico, un “subshift”. Así, estamos representando la dinámica de la hormiga, a travez de una estructura teórica única, y uniforme con respecto al grafo subyacente. Esto nos permite comparar las características de la dinámica de la hormiga en los distintos grafos, utilizando un language y herramientas de análisis comunes.

A continuación, enumeramos las propiedades de H_G , que fueron demostradas para los distintos grafos $G = \Gamma(k, d)$.

Transitividad *Es transitivo si y sólo si $d \geq 4$.* La demostración se basa en el hecho que la trayectoria de la hormiga no es nunca acotada.

Carácter Mezclador *Es mezclador si y sólo si $d \geq 4$ y $k \geq 4$.*

Entropía *Tiene siempre entropía positiva.*

Carácter Sáfico *Es sáfico si y sólo si $k = 3$.* Esta propiedad está asociada a la existencia de ciclos arbitrariamente largos.

Carácter Codificado *Es codificado si $d \geq 5$.* El hecho que la trayectoria de la hormiga esté confinada a un subgrafo con forma de árbol, hace que cada vez que la hormiga realiza un movimiento en zigzag, 2 veces consecutivas, no pueda volver a pasar por ninguna de las células que había visitado antes de hacer este movimiento.

STF *Es un subshift de tipo finito, si $d = 3$.* Cuando $k = 3$, la longitud del circuito más largo está acotada, esto significa que las células que la hormiga recorrió hace 8 iteraciones atrás, no serán visitadas nunca más en el futuro.

1.6 La hormiga como un sistema de cálculo

Otra manera de evaluar la complejidad de un sistema, es estudiar su capacidad de cálculo. Frecuentemente, nuestros modelos de cálculo son expresados como sistemas dinámicos discretos específicos (máquina de Turing, redes booleanas, autómatas celulares,...). Intuitivamente, tal sistema es complejo si es universal (desde el punto de vista del cálculo), en el sentido que puede “simular” toda máquina de Turing.

En la literatura se encuentran dos formas principales para mostrar la universalidad de un sistema:

- Si el sistema hace intervenir una noción de espacio isomorfo a $\mathbb{Z}$, la cinta de una máquina de Turing se interpreta directamente en este espacio [43, 31].
- Si el sistema actúa sobre un espacio $\mathbb{Z}^k (k \geq 2)$, es posible interpretar objetos más complejos, como un autómata y sus dos registros [32, 23], las puertas lógicas y las conexiones de un computador convencional [2, 24, 35].

Recientemente, [16] se ha hecho notar que si se sabe como simular conexiones y puertas lógicas, también se sabría simular la evolución de un autómata celular sobre $\mathbb{Z}$, a partir de una configuración finita. Además, como dicho autómata puede a su vez simular una máquina de Turing universal, se alcanzaría así la universalidad.

En el Capítulo 6⁴ nos colocamos en esta última perspectiva. Para comenzar, las puertas

⁴El trabajo correspondiente al Capítulo 6 fue desarrollado en colaboración con Andrés Moreira

lógicas, los alambres y cruces han sido simulados en grillas cuadradas y hexagonales. A partir de esto es posible construir cualquier circuito booleano finito. La Figura 1.13 ilustra algunas de las configuraciones obtenidas.

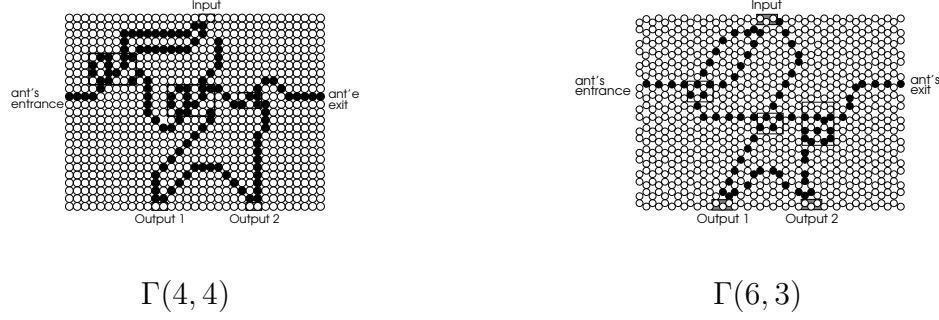


Figure 1.13: Duplicador de variables lógicas en las grillas cuadrada y hexagonal.

La idea consiste en ver la iteración de un autómata celular, sobre una configuración finita de tamaño n , como la acción de n copias de un mismo circuito booleano. Este nuevo objeto, puede ser representado como la configuración inicial para la hormiga, en que las células negras están restringidas a un rectángulo de altura independiente de n . La simulación de un autómata en todo instante, se obtiene apilando y conectando tales rectángulos. Se demuestra así, la universalidad de la hormiga de Langton para las grillas cuadrada y hexagonal (ver Figura 1.14).

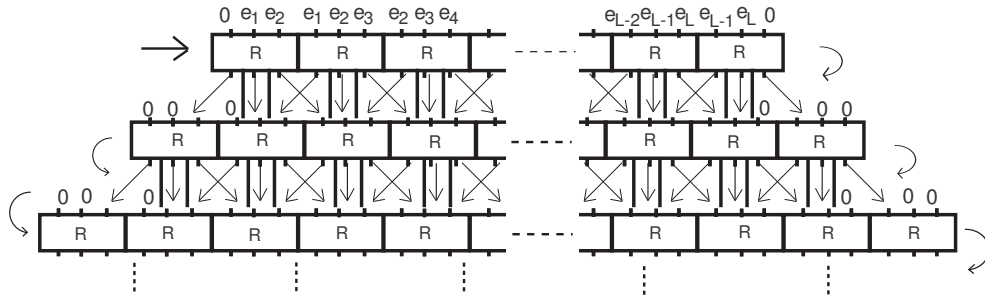


Figure 1.14: Configuración que simula un autómata celular.

Es importante observar que, al igual que en [17], las configuraciones iniciales son infinitas y periódicas, ya que no es posible prever el tiempo necesario para un cálculo. Los diferentes resultados de indecidibilidad en máquinas de Turing (problema del alto), se

traducen entonces, muy directamente, en resultados de indecidibilidad sobre la evolución de la hormiga, a partir de configuraciones infinitas (como, por ejemplo, la aparición de un motivo dado).

1.7 Conclusiones

Los diferentes análisis (combinatorial, de identificación como sistema dinámico, de capacidad de cálculo) ponen en evidencia los siguientes hechos:

- La dinámica de la hormiga de Langton es *muy difícil* de entender en las grillas cuadrada y hexagonal. Esto se aprecia en la existencia de problemas indecidibles asociados a su dinámica, al igual que en su capacidad de simular cualquier máquina de Turing.
- La dinámica de la hormiga es menos compleja cuando el grado del grafo es superior o igual a 5. Esto se expresa en el hecho que el lenguaje asociado a su trayectoria es *codificado*: su trayectoria puede descomponerse en trozos de trayectorias, los que pueden ser recombinados arbitrariamente.
- Si el grado del grafo es superior a 6 y las caras son triángulos, la trayectoria de la hormiga será simple. Será siempre la concatenación de dos ciclos sucesivos y de oscilaciones, en forma de zigzag, arbitrariamente largos.

Esta dicotomía ya se había puesto en evidencia en [25], donde se conjetura que hay un mecanismo de bloqueo cuando el ángulo de giro de la hormiga es mayor que 90° , lo que equivale a que el grado del grafo es superior a 4.

El método utilizado para demostrar la universalidad, muestra la existencia de una configuración periódica tal, que la hormiga no entra en una trayectoria regular.

Hemos demostrado (en el caso de grillas cuadrada y hexagonal) que existe una configuración inicial periódica tal que, si es modificada en un número finito de células, la hormiga puede simular una máquina de Turing. El mismo resultado no parece aplicarse, si se pone como condición que la configuración inicial simuladora sea finita, ya que ello estaría en contradicción con la conjetura que dice que la hormiga entra en una trayectoria regular cada vez que parte desde una configuración finita. Esta diferencia expresa (gracias a la simplicidad de la regla de la hormiga) que la complejidad de la tabla de

transición de una máquina universal, puede ser reemplazada por una complejidad en el espacio subyacente.

En los grafos $\Gamma(k, d)$ con $d \geq 5$, no podemos afirmar que el sistema no es universal si se permite partir de una configuración infinita. Cabe notar que la noción de periodicidad no puede ser definida en los grafos hiperbólicos, sin embargo, el hecho que la hormiga esté restringida a un subgrafo con forma de árbol, permite definir esta noción en dicho árbol. En base a esto, se podría tratar de demostrar un resultado de decidibilidad, empleando el mismo método que el utilizado en el Teorema 6.

Numerosos problemas permanecen abiertos para las grillas cuadradas y hexagonales. El hecho que la hormiga sea universal, hace suponer que la P-dureza es una cota holgada de la complejidad del problema y que es posible encontrar problemas NP-duros. En el caso de grado 3, se presentan los siguientes problemas:

1. Caracterización de las configuraciones que conducen a una evolución acotada.
2. Existencia de trayectorias regulares infinitas, sobre configuraciones iniciales finitas.
3. Si las trayectorias sobre configuraciones coloreables pueden ser acotadas.
4. Existencia de configuraciones coloreables en los grafos $\Gamma(2k + 1, 3)$

La respuesta al segundo problema es negativa para configuraciones finitas y coloreables, ya que la hormiga pasa un número infinito de veces por su punto de partida. Nosotros creemos que la respuesta es negativa, en general. Por otra parte, encontrar un método eficaz de búsqueda de configuraciones finitas que generan trayectorias regulares sería muy útil para encontrar una trayectoria diferente a la escalera, en el caso de la grilla cuadrada.

El caso de configuraciones coloreables es muy particular, y creemos que la trayectoria de la hormiga es necesariamente no acotada.

En el caso de los grafos $\Gamma(2k + 1, 3)$, aún cuando el Teorema 4 es verdadero, no se ha encontrado ninguna configuración coloreable y el teorema puede no ser válido más que en un conjunto vacío.

Parece razonable buscar nuevas nociones geométricas y combinatoriales (como las baldosas de Truchet), así como estudiar generalizaciones de la hormiga de Langton, para poner en evidencia la razón de la dificultad de su estudio sobre la grilla. Proponemos

considerar grafos subyacentes que se inyecten naturalmente sobre superficies orientables diferentes del plano. Nótese que la Definición 1 podría ser generalizada a cualquier grafo, fijando arbitrariamente una orientación para cada célula. Sin embargo, el análisis combinatorial de estos objetos tiene el riesgo de ser muy difícil, ya que no dispondríamos de herramientas geométricas.

1.8 Publicaciones

Esta tesis ha dado origen a los siguientes artículos, que fueron aceptados en revistas y conferencias internacionales: A. Gajardo, A. Moreira, E. Goles, “Complexity of Langton’s ant”, aceptado en *Discret Applied Mathematics*; A. Gajardo, A. Moreira, E. Goles, “Generalized Langton’s Ant: Dynamical Behavior and Complexity”, publicado en *STACS 2001, LNCS 2010*, (Springer-Verlag, 2001); y A. Moreira, A. Gajardo, E. Goles, “Dynamical behavior and complexity of Langton’s ant”, aceptado en *Complexity*.

Además, se encuentran en preparación dos trabajos más; el primero de ellos en relación con el Capítulo 3 (conjuntamente con J. Mazoyer); el segundo de ellos relacionado con el Capítulo 7 (conjuntamente con E. Goles).

Chapter 2

Definitions and basic notions

Here we give the definition of the ant rule and some notations that will be used the long of the text. We also recall some notions of the graph theory.

2.1 The ant's rule

Notation: Since the movement of the ant consist in turning to the left and to the right, we will denote the cell's states *to-left* and *to-right* instead of *white* and *black*.

Definition 1. *Let us consider a representation of a planar undirected graph $G = (V, E)$ on $\mathbb{R}^2$. At a given instant t , each cell v is in a state $c_t(v)$ defined by the function $c_t : V \rightarrow \{\text{to-left}, \text{to-right}\}$. The ant is at position $h_t = (i, j)$, over an edge $\{i, j\}$ (heading to the cell j). Let us consider a uniform orientation of the plane to well define the left (clockwise) and right (counterclockwise).¹ Let $\{j, l\}$ be the edge that is to the left of $\{i, j\}$, with respect to j and $\{j, r\}$ the edge that is to the right of $\{i, j\}$. The dynamics of the system is the following:*

$$h_{t+1} = \begin{cases} (j, l) & \text{if } c_t(j) = \text{to-left} \\ (j, r) & \text{if } c_t(j) = \text{to-right} \end{cases}$$
$$c_{t+1}(j) = \begin{cases} \text{to-right} & \text{if } c_t(j) = \text{to-left} \\ \text{to-left} & \text{if } c_t(j) = \text{to-right} \end{cases}$$

¹Since $\mathbb{R}^2$ is an orientable surface, the concepts to right and left are well defined.

The function c_t is called the configuration of the graph at time t .

Figure 2.1 shows an example of the definition of edges $\{j, l\}$ and $\{j, r\}$. A *constant* configuration is a configuration that assigns the same state to all the cells. We say that a configuration is *finite* if it differs from a constant configuration only in a finite number of cells.

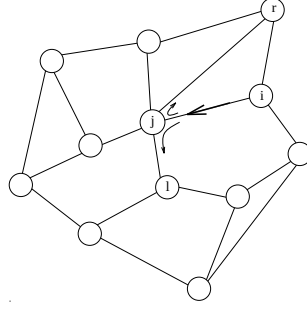


Figure 2.1: The ant turns the most possible to the left or to the right.

Given a configuration c_0 and an initial position $h_0 = (i_0, j_0)$ let us define the ant position at time t as $(i_t, j_t) = h_t$. We say that the infinite sequence of cells $(i_0, j_0, i_1, j_1, i_2, j_2, \dots)$ is the *ant's trajectory* generated by (c_0, h_0) . Conversely we say that a sequence of cells is an ant's trajectory if there is an initial configuration and a position that generate it.

Keeping the former notation, a finite sequence $(a_n)_{n=0}^N$ is called *ant-path* if there is a configuration and a position (c_0, h_0) such that $(i_0, j_0, i_1, j_1, \dots, i_{N-1}, j_{N-1}) = (a_n)_{n=0}^N$. In that way, an ant-cycle is an ant-path such that $a_0 = a_N$. A simple ant-path is an ant-path that does not repeat any cell and a simple ant-cycle is an ant-cycle such that only the first cell is repeated and it is repeated only one time.

We wish to distinguish a special path that is in general an ant-path. Given the representation of a graph G , a *zigzagging path* $(a_0, a_1, \dots, a_N)$ is an ant path such that the pair (c_0, h_0) which generate it satisfies:

$$c_0(a_n) = \begin{cases} \text{to-left} & \text{if } c_0(a_{n-1}) = \text{to-right} \\ \text{to-right} & \text{if } c_0(a_{n-1}) = \text{to-left} \end{cases}$$

for any n between 1 and N .

In other words it is a path such that the ant turns alternately to the left and to the right.

We say that an ant's trajectory $(a_n)_{n \in \mathbb{N}}$ is *eventually regular* of period p if there is a time T such that for any $t \geq T$ $c_{t+p}(a_{t+p}) = c_t(a_t)$. That is, after a time T the ant repeats periodically the same movement. This may be because the ant's dynamics is periodic or because the ant propagates regularly in some direction as when starting with a constant configuration in the square grid (Introduction).

An ant's trajectory is said to be *bounded* if it is composed by a finite number of cells. In that case, it is easy to see that the trajectory is *periodic*, i. e., $(\exists p \in \mathbb{N})(\forall n \in \mathbb{N}) a_{n+p} = a_n$.

Remark 1. *A very important property of the rule is its **reversibility**. It is easy to see that we can always know where was the ant and what was the state of each cell in the previous iteration. In fact it is enough to invert the direction of the ant, to apply the rule, and to invert once more the direction of the ant. In that way we obtain the previous configuration and the previous position of the ant.*

2.2 Graphs

An undirected simple *graph* is a pair (V, E) where V is the set of *vertices*, and E is a set composed by no-ordered pairs of elements of V (subsets of V of cardinality 2). The elements of E are called *edges* and they are denoted by $\{v, u\}$.

In this thesis principally we work with undirected simple graphs so we will simply call them *graphs*. They are used as the underlying space for the ant; so, we will prefer to call the vertices *cells*.

A *directed graph* is a pair (V, E) where V is the set of *vertices*, and E is a family $\{e_i\}_{i=1}^n$ of elements of $V \times V$, called *arcs*. An arc can appear more than once in E .

Two arcs (or two edges) are *adjacent* if and only if they have one endpoint in common. Two cells v and u are adjacent if (u, v) or (v, u) is in E ($\{v, u\} \in E$). An arc (an edge) is *incident* to a cell u if u is one of its endpoints. A *path* D is a sequence of cells $(u_0, u_1, \dots, u_k)$, such that $(u_i, u_{i+1}) \in E$ ($\{u_i, u_{i+1}\} \in E$) for all i . We call k the *length* of D and we say that the distance between u_0 and u_k is less or equal than k . A *cycle* C is a path whose extreme cells coincide. We say that a path (a cycle) is *simple* if it does not repeat cells (other than the extreme cells, in the case of cycles). A directed

graph is *strongly connected* if for all v_0, v_f in V there is a path $(u_0, u_1, \dots, u_k)$ such that $u_0 = v_0$ and $u_k = v_f$. An undirected graph satisfying the same condition is simply called *connected*.

For undirected graphs, we define: The *neighbors* of a cell u are the cells $N(u) = \{v \in U : \{u, v\} \in E\}$. The *degree* of u is $|N(u)|$. A graph where all the cells have the same degree is said to be *regular*. A *leaf* is a cell with degree 1. A graph is *bipartite* if V can be partitioned into two sets W and U such there are not adjacent cells in the same set.

A graph is said to be *planar* if it may be injected in $\mathbb{R}^2$, the cells being represented by points and the edges by simple curves, in such a way that the curves *do not intersect*. A planar graph is *locally finite* if it may be injected in $\mathbb{R}^2$ such that any sphere of finite radius in $\mathbb{R}^2$ contains a finite number of nodes. In a planar graph, a *face* is one of the regions of the partition induced by the graph. The *cardinality* of a face is the number of edges in the polygon that defines it. The *dual graph* of G , G' , is defined as the graph $G' = (U', E')$, where U' is the set of faces of G , and $(i', j') \in E'$ if and only if i' and j' have a common edge.

Chapter 3

Some particular combinatorial properties

In this chapter we show some combinatorial properties that have validity on general planar graphs satisfying only some regularity conditions. They concern even degree graphs and graphs of degree three. Proofs are based on notions that were introduced in previous researches on the ant. Particularly the notion of *corner* [5] and *Truchet tile* [39].

3.1 Graphs of even degree

This section generalizes a Bunimovich-Troubetzkoy theorem and establishes some additional related facts. The concerned theorem is the most important one about the Langton's ant. The idea of the proof can be used in similar theorems over the line and other graphs as well as in some generalizations of the ant's rule.

Theorem 1. [6] *In the square lattice, an ant's trajectory is always unbounded.*

Proof. Let c be a configuration on the square grid. Let us label the cells with colors like a checkerboard. Since there are not adjacent cells with the same color, the ant visits alternately the black and white cells. Also, its position is alternately horizontal and vertical. If the ant starts in horizontal position pointing to a black cell, it will be always in horizontal position when pointing black cells and in vertical position when pointing to white cells.

Suppose that the ant stays in a bounded region. Consider the set of cells Ω which the ant visits infinitely often. Consider the rightmost set of cells in Ω and the topmost cell (i_{max}, j_{max}) in that set. Let us suppose (i_{max}, j_{max}) is black. Since the ant visits this cell infinitely often, the ant finds the cell alternatively in *to-left* state and in *to-right* state. When it finds the cell in *to-left* state the ant goes to $(i_{max}, j_{max} + 1)$. That occurs infinitely often, then there exists a cell in Ω that is strictly up to the cell (i_{max}, j_{max}) , which is a contradiction.

□

Using the same idea, a stronger fact was remarked by S. Troubetzkoy:

Theorem 2. [44] *In the square lattice the set of cells that the ant visits infinitely many times has no corners.*

These theorems are based on the checkerboard coloring of the grid, but this is not an exclusivity of the square lattice. In fact, to color as a checkerboard is to divide the faces into two sets: the black ones and the white ones in such a way that no adjacent faces are in the same set. In graph theory, this property express exactly that the *dual* graph is *bipartite* (see Section 2.2 for definitions). And a necessary and sufficient condition for a planar graph to have a bipartite dual graph is that all its cells have even degree [9].

This coloring property allows to prove that, for a given initial position of the ant, a fixed orientation is associated to each edge, so that the ant passes over it always in that direction. This property is strong and it implies several properties that are commented here. In particular Theorem 1 is true on the infinite $\Gamma(k, 2d)$ graphs.

Lemma 1. *If the dual of $G = (V, E)$, say $G' = (V', E')$, is bipartite, with $V' = W \cup B$, W and B verifying that there are not adjacent cells in the same set. Then, the faces corresponding to W will remain always at the same side of the ant.*

Proof. Let us suppose that at the beginning the face at the left of the ant belongs to W . The face at the right of the ant is in B .

In the next step the ant will turn around a face. If it turns around the left face, this one will remain at its left, and another face in B will appear at its right. If it turns to the right, the face in B will remain to its right, and a new face of W will appear at its left.

□

Lemma 1 has the direct but strong consequence that the ant passes over each edge always in the same direction. A first consequence of this has relation with the Truchet tiles. Given a cell, the ant can arrive to it only through certain edges, which alternates with the edges through which the ant can exit it. In the language of Truchet tiles, that means that a contour will neither intersect nor touch itself. For example, in the triangular grid, if the ant begins pointing down, the ant can only arrive to the cells from the upper edge and from the left and right bottom edges. A coloring is shown in Figure 3.1. It assures that a contour is colored with a uniform color and that no contour of the same color touches each other.

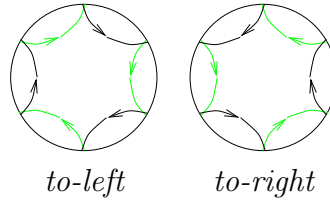


Figure 3.1: The coloration of a Truchet tile of degree 6, the color changes as a function of the cell state.

In a graph of odd degree, two colors are not enough to obtain this property. If we want to color with only two colors, we are forced to alternate the color of the arrows in a Truchet tile. Since the degree is odd, we will finish with two neighboring arrows of the same color. Hence, at least three colors are necessary.

Definition 2. Let $G = (V, E)$ be an undirected graph and $S = (U, D)$ a subgraph ($U \subset V$, $D \subset E$). We say that a cell $u \in U$ is a corner of S if and only if its degree in G is strictly greater than its degree in S and its degree in S is 0, 1, or 2. Additionally, if its degree in S is 2, these 2 incident edges belong to the same face of G (see Figure 3.2).

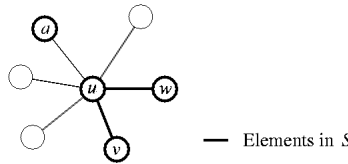


Figure 3.2: u is a corner of S .

Lemma 2. *Let G be a graph such that the ant passes over each edge always in the same direction. Then, the subgraph S_i of cells and edges that the ant visits infinitely often have no corners.*

Proof. Let us remark first that a corner in S_i has degree at least 2 in G , because the degree of any cell in S_i is at least 1. Then a corner in S_i has degree at least 2 in S_i , because the input edge is different from the output edge. Let us suppose that S_i contains a corner u . Let v and w be the unique neighbors of u belonging to S_i (see Figure 3.3). Without loss of generality, let us suppose that the ant passes over the edge $\{v, u\}$ pointing always to u . The ant can visit u only from v . A half of the times that the ant visits u , it turns to the left. Then, an infinity of times, the ant goes over an edge that does not belong to S_i . This is a contradiction. $\square$

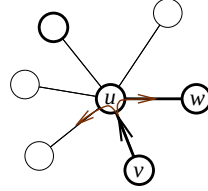


Figure 3.3: The ant exits from S_i an infinite number of times.

In the square grid, for example, a set without corners is an infinite row or a union of infinite rows. No set without corners is finite. Also, the proof of Lemma 2 says something stronger about S_i : It says that the degree of the cells in S_i is equal to its degree in G or it is greater than 3. Another property can be concluded:

Proposition 1. *Let G be is a graph whose cells have degree at least 3 and such that the ant passes over each edge always in the same direction. Considering the evolution of the ant in forward and backward time, let us define T as the graph of cells and edges that are visited by the ant. Then, a cell is visited exactly one time if and only if it is a corner of T .*

Remark 2. *If $G = \Gamma(k, d)$ is an infinite bi-regular graph with $k \geq 4$, any finite subgraph has at least one corner [38].*

The following generalization of Theorem 1 is a direct conclusion from Lemmas 1 and 2 and Remark 2.

Theorem 3. *Let $G = \Gamma(k, d)$ be an infinite bi-regular graph, with $k \geq 4$, such that the ant passes over each edge always in a fixed direction that depends only on the initial position of the ant. Then the ant's trajectory is unbounded for any initial configuration.*

The hypotheses of the last theorem are difficult to verify but the contrapositive is more interesting: *in a bounded trajectory over an infinite bi-regular graph $G = \Gamma(k, d)$, with $k \geq 4$, there must be an edge that is traveled through in both directions.* Moreover, from the proof of Lemma 2, we can specify that every edge incident to a corner of S_i is traveled through in both directions.

The following corollary concludes this section.

Corollary 1. *The ant's trajectory is unbounded over the $\Gamma(k, 2d)$ graphs, with $k \geq 4$, for any initial configuration.*

3.2 Graphs of degree 3

The behavior of the ant within the graphs of degree three may be bounded (periodic) or not: there exist many initial configurations for which the ant has a bounded trajectory, and there are also configurations which leads to an unbounded trajectory. Figure 3.4(a) shows a configuration in an arbitrary bi-regular graph of degree 3; this one and 3.4(b) produce a bounded behavior. Figure 3.4(c) shows a configuration for which the ant has an unbounded behavior. A similar configuration can be defined in any bi-regular graph so as to have an unbounded trajectory.

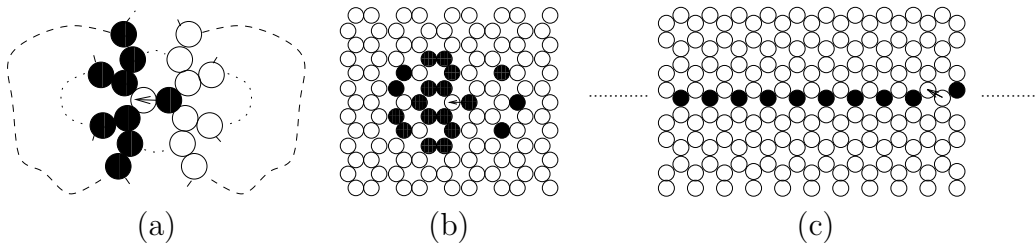


Figure 3.4: (a) Initial configuration which leads to a bounded behavior on a general $\Gamma(k, d)$ graph. Dashed lines stand for simple paths that do not intersect the rest of the configuration. Dotted lines stand for a face. (b) An implementation of (a) in the hexagonal grid. (c) An initial configuration which leads to an unbounded behavior.

Bounded trajectories have frequently a structure that F. Wang [46] calls *reflector*. This consists in a trajectory that passes two times over the same edge (say time t_1 and t_2) in opposed directions. If the trajectory that the ant has followed before t_1 does not intersect the trajectory between t_1 and t_2 , the trajectory after t_2 will be exactly the reverse of the one before t_1 because the ant is its own inverse. The trajectory between t_1 and t_2 is called reflector. By replacing the trajectory before iteration t_1 by another reflector that does not intersect the first one, we obtain bounded trajectories. Unfortunately, Wang also shows bounded trajectories that do not contain reflectors. The configuration of Figure 3.4(b) is composed by two identical reflectors, one in front of the other. The ant is exactly on the edge that connect them. This one is the smallest configuration leading to a bounded behavior in the hexagonal grid..

In the case of the hexagonal grid (and all the $\Gamma(2k, 3)$ graphs), when starting with a configuration where all the cells are in the same state, the ant generates configurations that have bilateral symmetry. See Figure 3.5 for a simulation of the ant over $\Gamma(6, 3)$. This behavior is common to a large class of initial configurations. This chapter is dedicated to prove a theorem that explains this. We use the notion of Truchet tiles described in Section 1.4.2.

As we said in that section, there exist a direct relation between configurations and tilings with Truchet tiles. Figure 3.6 shows how a configuration in $\Gamma(6, 3)$ is transformed into a tiling with Truchet tiles. To emphasize the idea of tiling, one can draw the Truchet tiles triangle shaped.

We will forget -for a moment- the ant, to study the relation between configurations and tilings.

Let us consider now that the arrows have a color, and that the condition for two tiles to be adjacent is that the color is conserved in the continuity of the arrows. For two colors, the total set of tiles is: $\{ \text{⬇}, \text{↘}, \text{↙}, \text{↗}, \text{↖}, \text{↕}, \text{↗}, \text{↖} \}$.

For each tiling t there is an associated configuration $\mathcal{E}(t)$ which is obtained by replacing the left-oriented tiles by a cell in *to-left* state and the right-oriented tiles by a cell in *to-right* state. A configuration may be associated to many tilings; Figure 3.7 shows several tilings whose associated configuration is the same, the difference lies on the coloring of the contours.

As the arrows indicates the ant's movement in one time step, the contours indicates the ant's movement in several time steps. The ant follows a contour but it modifies it at the same time. Modified tiles are those that the ant has already visited. Then given a

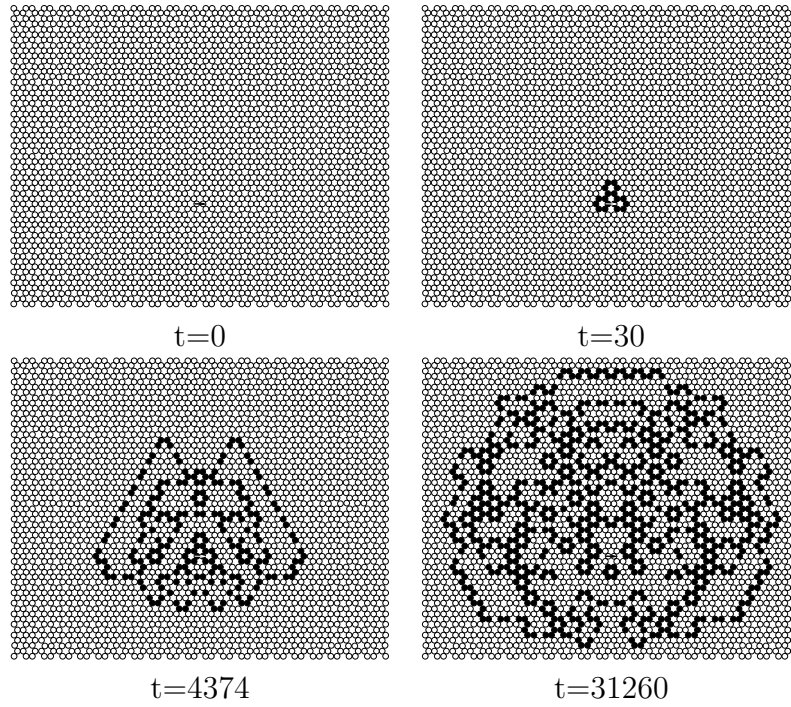


Figure 3.5: The evolution of the ant over the hexagonal grid starting with all the cells in *to-left* state.

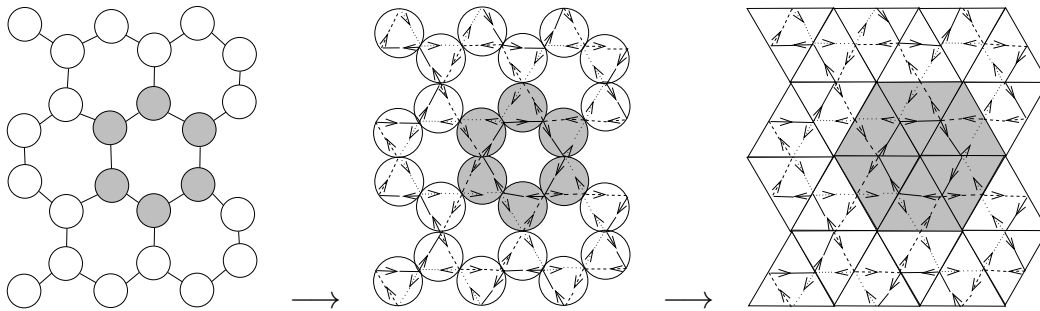


Figure 3.6: The cells are replaced by Truchet tiles.

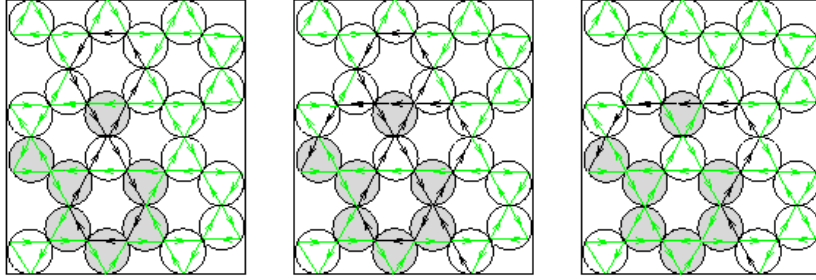


Figure 3.7: Different tilings for the same configuration.

contour, we are sure that the ant will follow it until some cell appear for a second time in the contour.

Given an initial configuration, we can predict the ant position in several time steps (at least k) just by following the principal contour (the contour on which the ant is, see Section 1.4.2).

A simple contour is a contour, closed (finite) or not (infinite), that passes at most one time by each cell. So, if at some iteration the ant is over a simple contour, we know that it will follow it entirely. Therefore we are interested in characterizing and studying the configurations whose contours are simple. The condition of simplicity of contours can be translated as a restriction in the tile set as follows. Let τ' be a set of colored Truchet tiles whose three arrows have a different color. In a tiling, the arrows of a contour have all the same color. If a contour passes two or more times by a cell, the Truchet tile associated to this cell must contain two arrows of the same color, then there cannot exist a tiling with τ' whose associated configuration have no simple contours.

Let τ_q be the set of all the colored Truchet tiles satisfying that the three arrows in a tile have different colors, and using only q colors. For example, τ_2 is an empty set. Let C_q be the set of configurations that are associated to some tiling with τ_q . C_q grows with q . If an infinite number of colors is allowed (τ_∞), C_∞ contains all the configurations whose contours are simple. If q is finite, C_q may be strictly contained in C_∞ . For example, configuration where all the cells are in the same state satisfies that all the contours are simple but it does not belongs to C_3 in the case of the $\Gamma(2k+1, 3)$ graphs.

The simplest case is for $q = 3$; $\tau_3 = \{ \text{⦿}, \text{⦿}, \text{⦿}, \text{⦿} \}$. This case is very interesting for several reasons. The first one is that in general a single configuration may be associated

to infinite many tilings. When $q = 3$ there are only three tilings associated to each configuration. In fact, defining the coloring of a given tile, impose the coloring of the three adjacent tiles. So, it imposes by recursion the coloring of all the tiles in the graph.

A second reason is that the problem of tiling with τ_3 is related to another more natural tiling problem: $T = \{ \blacktriangle \}$. A solution of τ_3 defines a solution of T , in fact, to go from a tiling with τ_3 to a tiling with T it is enough to make the transformation described in Figure 3.8. Solid lines are associated with black color, dashed lines with gray color and dotted lines with light gray color.

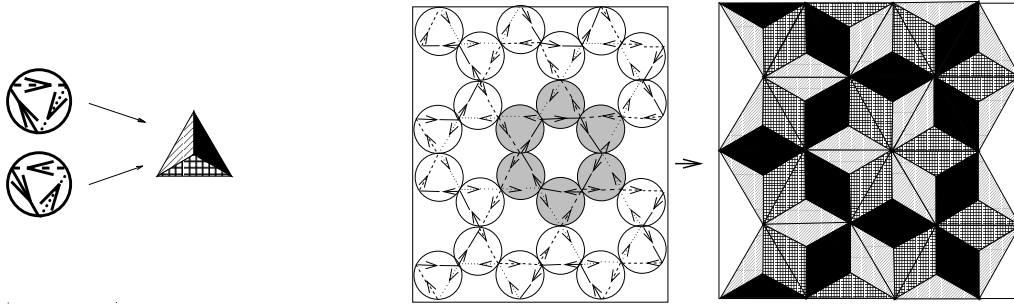


Figure 3.8: Transforming a tiling with τ_3 into a tiling with T .

The problem to know whether a bounded region can be tiled or not with T , given that the boundary tiles are fixed seems to be approachable by using tilings groups [11]. But the existence of non local transformations avoids to apply the classical methods to prove that it is solvable in polynomial time [41]. As yet, the problem remains solvable only in exponential time.

The biggest interest of set τ_3 is related to the ant's dynamics. We prove the property enunciated at the beginning of this chapter precisely by using this set of tiles.

For a given configuration in C_3 , we call **correct coloring** a tiling to which the configuration is associated, and a free assignation of colors to the arrows inside the Truchet tiles is called **coloring**.

Theorem 4. *Let c be a configuration in C_3 for a regular graph G of degree three. Then, if the ant starts over a finite contour, after a complete tour of the ant over this contour, the new configuration is also in C_3 .*

Proof. Since c is in C_3 , all the contours are simple, including the principal contour. Let us suppose that c is colored with the colors green, blue, and red, and that the principal

contour is red. If the principal contour is closed, the ant will come back to the origin. Let c' be the configuration after a complete tour of the ant over the principal contour. The ant passes one and only one time over each cell on the principal contour. When the ant passes over the cells, the arrows on these cells change their orientation, this includes each arrow of the principal contour. Thus, the principal contour of c is also a contour in c' but with the opposite orientation. To prove the theorem we will define a coloring for c' as follows:

- all the cells not crossed by the principal contour are colored as in c .
- the principal contour of c remains red.
- the arrows in the cells crossed by the principal contour, that are not in the principal contour, change their color from blue to green and from green to blue according to the coloring of c .

The new coloring is correct, as Figure 3.9 shows. We just must verify the correctness of the coloring in the cells that have changed. Figure 3.9 shows a cell a colored in the configuration c and in the configuration c' . Without loss of generality we assign colors to its arrows in the coloring of c . Since there is a green arrow heading to its *unique* neighbor u outside the principal contour, we know that in u there is a green arrow that leaves the cell a . We also know that in u there is a blue arrow heading to a . After the passage of the ant, all the arrows in a and also in l and r change their orientation. Then to preserve the correctness of the coloring it is sufficient to interchange the colors of the green and blue arrows inside the cells of the principal contour. The same thing happens in the cells l and r that belong to the principal contour, and thus the correctness of the coloring of c' is proved.

If u belongs to the principal contour the analysis does not change, since, by interchanging the colors of the arrows, we respect the colors of the exiting and entering arrows. $\square$

The movement of the ant over a finite contour defines a transformation in the set of tilings with T .

If the ant reaches an infinite contour, it escapes and never returns to the initial cell. If the initial configuration is in C_3 and does not contain infinite contours, then they will never appear, since at each tour the ant changes only finitely many cells. Thus, the ant will return to the initial cell an infinite number of times, following always simple trajectories. This assures that no stairway will ever appear. Periodic trajectories might appear, but no one has been found yet over initial configurations in C_3 .

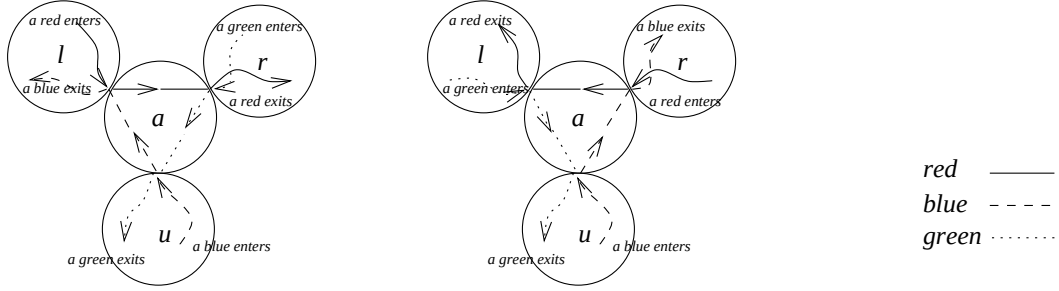


Figure 3.9: The coloring of c' .

Let us remark that using exactly three colors is the key in this theorem. An initial configuration where all the contours are simple, but that is not in C_3 , may lose this property after a tour of the ant over one of its contours. It is the case of the graph $\Gamma(7, 3)$ and the configuration where all the cells are in the same state. This configuration is associated to a tiling with τ_4 . After a tour of the ant, the new configuration contains no simple contours (see Figure 3.10). The reason is very simple: when we define the coloring of the new configuration in the proof of the theorem, we interchange the color of the arrows that are not red, this preserves the correctness of the coloring because we know that the colors of the neighboring cells are the same. If there are 4 colors, and we know 2 of them inside a cell, we cannot infer the third.

The analogous theorem is not true in a graph with a larger degree. It is enough to regard the ant evolution over a configuration where all the cells are in the same state. When the degree of the graph is larger than 3, after a tour of the ant, contours passing two times by the same cell appears (see Figure 3.11). The degree is as fundamental in this theorem that even the condition of planarity of the graph can be eliminated and the result remains true. Nevertheless, for certain generalizations of the ant's rule in the square lattice, there is a similar result [21].

Theorem 4 provides the proof for the symmetrical behavior of the ant over the configurations where all the cells are in the same state: In the hexagonal grid and in all the $\Gamma(2k, 3)$ graphs, the configuration where all the cells are in the same state is in C_3 and if we consider one of the associated tilings together with the ant in some initial position, the picture results to be symmetric under reflection with respect to the axis that is perpendicular to the ant position¹ (if we do not consider the orientations of the

¹To talk properly of symmetry under reflection in the hyperbolic graphs, we consider a representation of the graph in the plane being $2k$ -symmetrical with respect to a point at the center of one of the faces of the graph. We suppose also that the ant is over one of the edges of this face.

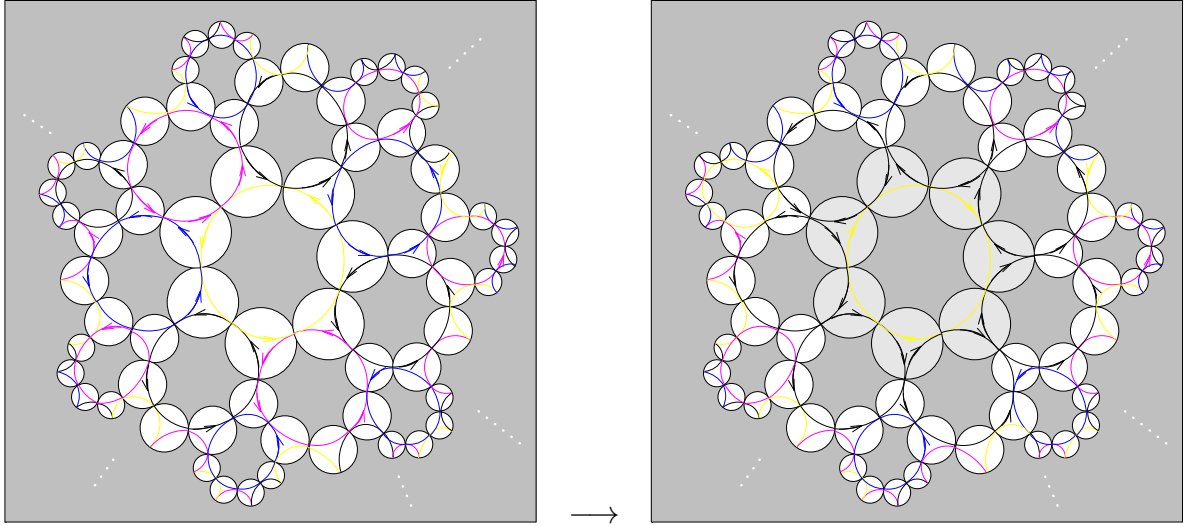


Figure 3.10: The theorem does not work if we admit more than three colors. In the figure all the cells begins in *to-left* state. If the ant travel through the central contour (yellow colored), a not simple contour appears (black colored).

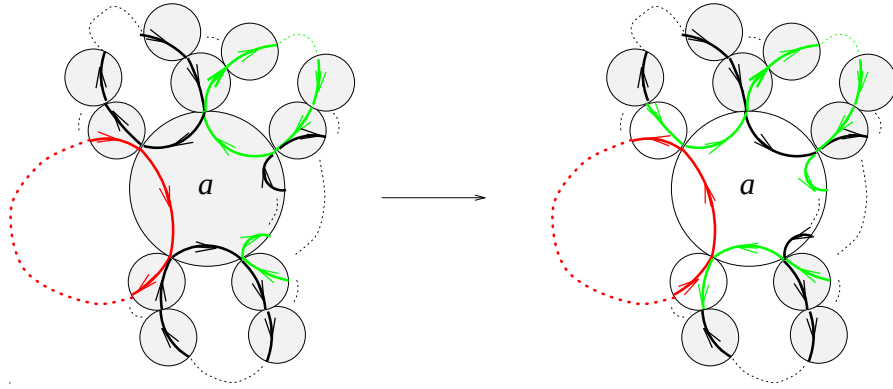


Figure 3.11: The theorem does not work if the degree of the graph is larger than three. In the figure the cells are represented by circles and the contours are represented by wide curbs bold arrows. When all the cells are in *to-right* state, the ant follows a face (red contour), and the resulting configuration contains a contour (green colored) that passes two times by the cell a .

arrows). Each tour of the ant has this symmetry, so at each tour the ant modifies the space symmetrically. In that way, symmetry is preserved.

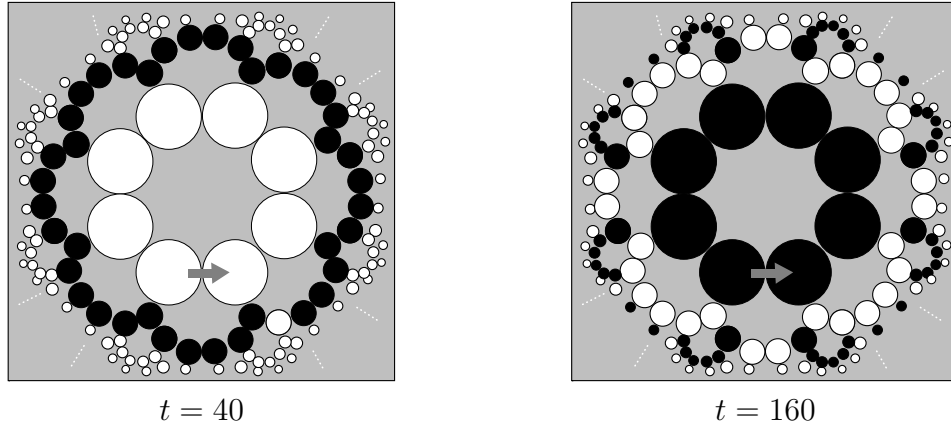


Figure 3.12: Evolution of the ant on the graph $\Gamma(8, 3)$ starting with all the cells in *to-left* state.

Chapter 4

$\Gamma(k, d)$ graphs, with $d \geq 5$

When the underlying graph has degree larger or equal than 5, the ant cannot reach all the cells of the graph, given a fixed starting position. To get a feeling of this, try to draw a simple ant path in the triangular grid; you will easily see that the unique simple path is a zigzagging line and that the unique simple cycle is a triangle.

For most of the following results, we will consider general planar graphs satisfying (H):

A graph is said to verify the hypothesis (H), if all its cells have degree $\geq d$, and all its faces have cardinality k , with $d = 5$ and $k \geq 4$, or $d \geq 6$ and $k \geq 3$.

Our first proposition says that there are no other simple ant cycles than the faces. This fact can be true only if there is a limit in the “curvature” of the simple ant-paths. They are a kind of straight lines. Let us recall that in hyperbolic geometry, there exists an infinity of parallel lines that go in different directions. Proposition 3 shows exactly this: the ant can go to no one of the cells that are “behind” it. In other words: given an initial position of the ant, there is a complete hemisphere that cannot be reached through simple paths.

Our second result shows that the simple paths form a tree. Starting from a given position, they do not intersect each other; they do not form cycles, which would contradict the first proposition. Such a tree does not contain all the cells of the graph, there is an infinity of cells that cannot be reached through a simple ant-path. To conclude, we prove that general ant-paths do not go farther than simple ant-paths: a cell that can be

reached through an arbitrary ant-path can also be reached through a simple ant-path. This proves that, in fact, the ant is confined to the tree defined by the simple ant-paths.

In Section 4.2 we prove that the ant's trajectory is eventually regular if the starting configuration is finite.

4.1 Main result

The proof of the following proposition uses techniques from [37].

Proposition 2. *Let G be a graph satisfying (H). Then, the unique simple ant-cycles are the faces.*

Proof. We will prove this proposition by contradiction. Let P be a simple ant-cycle, and suppose that P is not a face.

n = the cycle's length,

m = number of enclosed cells (excluding the cells into the cycle),

e = number of enclosed edges (including the edges into the cycle).

f = number of edges connecting the cells in P with the enclosed cells.

Then

$$e = \frac{md}{2} + n + \frac{f}{2}$$

and

$$f \geq (d-2) \left\lfloor \frac{n}{(k-1)} \right\rfloor$$

because for each $k-1$ consecutive cells in the cycle, there must be at least one on which the ant turns to the outside of the cycle, thus adding $d-2$ edges. We take the integer part of this because the initial cell can admit a number of enclosed edges different from 0 or $d-2$.

From Euler theorem we know that $e = n + m - 1 + \frac{n+2m-2}{k-2}$, then,

$$\begin{aligned}
n + m - 1 + \frac{n + 2m - 2}{k - 2} &\geq \frac{md}{2} + n + \frac{d - 2}{2} \left\lfloor \frac{n}{k - 1} \right\rfloor \\
\frac{n + 2m - 2}{k - 2} - 1 &\geq \frac{m(d - 2)}{2} + \frac{(d - 2)(n - k + 1)}{2(k - 1)} \\
n - k &\geq \frac{m\gamma}{2} + \frac{(\gamma + 4)(n - k + 1)}{2(k - 1)} \\
2(n - k)(k - 1) &\geq m\gamma(k - 1) + (n - k + 1)(\gamma + 4) \\
&\geq \gamma(m(k - 1) + n - k + 1) + 4n - 4(k - 1) \\
(n - k)(2k - 6) - 4 &\geq \gamma(m(k - 1) + n - k + 1) \\
(n - k)(2k - 6) - 4 &\geq \gamma(m(k - 1)) + \gamma(n - k) + \gamma
\end{aligned}$$

Where $\gamma = (d - 2)(k - 2) - 4$.

The last equation carries a contradiction if and only if $d \geq 5$ and the graph is infinite. To see this, we analyze two cases: $d = 5$ and $k \geq 4$; $d = 6$ and $k \geq 3$.

$d = 5$ and $k \geq 4$: $\gamma = 3k - 10$

$$\begin{aligned}
(n - k)(2k - 6) - 4 &\geq \gamma(m(k - 1)) + (3k - 10)(n - k) + 3k - 10 \\
-(n - k)(k - 4) &\geq \gamma(m(k - 1)) + 3k - 6 \\
-(n - k)(k - 4) - 3(k - 2) &\geq \gamma(m(k - 1)) \\
&\geq 0
\end{aligned}$$

$d = 6$ and $k \geq 3$: $\gamma = 4(k - 3)$, then,

$$\begin{aligned}
(n-k)(2k-6)-4 &\geq \gamma(m(k-1)) + 4(k-3)(n-k) + 4(k-3) \\
-2(n-k)(k-3) &\geq \gamma(m(k-1)) + 4(k-2) \\
-2(n-k)(k-3)-4(k-2) &\geq \gamma(m(k-1)) \\
&\geq 0
\end{aligned}$$

This prove the proposition. When (H) is not satisfied, we can verify that no contradiction appears:

$d = 5$ and $k = 3$: $\gamma = -1$, then,

$$\begin{aligned}
(n-3)(6-6)-4 &\geq \gamma(m(3-1)) - (n-3) - 1 \\
n-6 &\geq -2m
\end{aligned}$$

$d \leq 4$ and $k \geq 3$: $\gamma \geq d-6$, then,

$$\begin{aligned}
(n-k)(2k-6)-4 &\geq \gamma(m(k-1)) + (d-6)(n-k) + (d-6) \\
(n-k)(2k-d)+2 &\geq \gamma(m(k-1)) + d \\
2(n-k)(k-2)+2 &\geq 2m(d-6) + d \\
2(n-k)(k-2)+2 &\geq -(2m+1)(6-d) + 6
\end{aligned}$$

□

When G does not satisfy (H) we can find several examples showing that Proposition 2 is not true, i. e., G allows ant cycles different from the faces (see Figure 4.1).

Definition 3. *Given a graph of degree ≥ 5 and an ant position (a, b) , we define a set of cells $\{a_i\}_{i \in \mathbb{Z}}$ recursively as follows.*

- $a_0 = a$

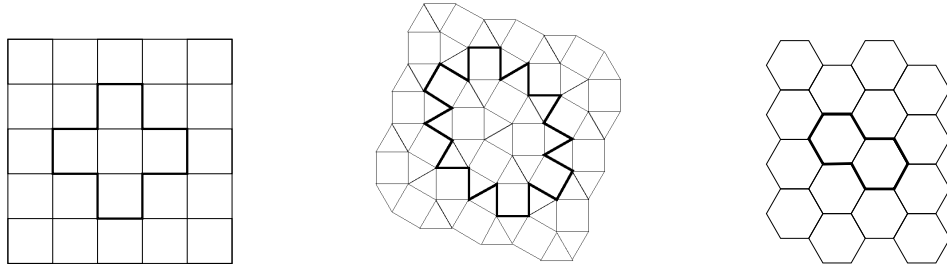


Figure 4.1: Some examples of graphs that do not satisfy (H) and allows ant cycles different from the faces.

- a_1 and a_{-1} as the neighbors of a_0 that are respectively to the left and to the right of b .
- if $a_i, a_{i-1}, \dots, a_{-i}$ are defined, let us define a_{i+1} as the 3th neighbor of a_i found starting at a_{i-1} and going counterclockwise. Analogously, define a_{-i-1} as the 3th neighbor of a_{-i} found starting at a_{-i+1} and going clockwise.

The cells $\{a_i\}_{i \in \mathbb{Z}}$ are called **boundary cells** with respect to (a, b) .

We define the **shaded zone** of the graph with respect to (a, b) as the set of cells delimited by the boundary cells with respect to (a, b) that contains the cell b and including the boundary cells themselves.

The boundary cells are defined such that for each one of them (with exception of a_0), exactly 4 neighbors are inside the shaded zone. Since the graph shall have degree ≥ 5 , for each boundary cell there are at least one neighbor outside the shaded zone. Figure 4.2 shows a shaded zone in a $\Gamma(4, 5)$ graph. Let us remark that the boundary cells are defined independently of the cardinality of the graph faces and the graph degree.

Proposition 3. *Let us consider a $\Gamma(k, 5)$ graph with $k \geq 4$. Then, no simple ant-path starting with the cells (a_0, b) may exit the shaded zone of $\Gamma(k, 5)$ with respect to (a_0, b) .*

Proof. To exit the shaded zone, the ant must do it through one of the boundary cells. Let O be a simple ant-path exiting the shaded zone.

Assertion: *The first time the ant visits a boundary cell it visits a_1 or a_{-1} , and it does it through the two incident edges that are strictly inside the shaded zone.*

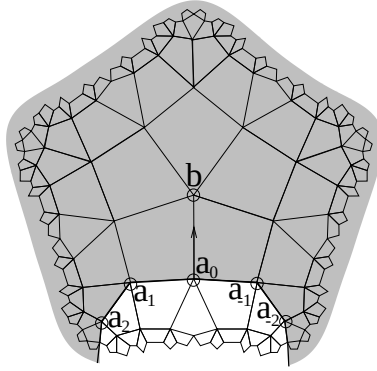


Figure 4.2: The shaded zone and the boundary cells with respect to (a_0, b)

Let t be the first time that the ant visits the boundary, $O_t = a_i$ for some $i \in \mathbb{Z}$. $O_0, \dots, O_{t-1}$ are strictly inside the shaded zone. Figure 4.3(a) shows a simple ant-path (A) that visits all the boundary cells to the left of a_0 without entering the shaded zone. Let us suppose without loss of generality that $i > 0$ and let us consider the path $(A, O_0, \dots, O_{t-1})$. It is a simple ant-path and $(A, O_0, \dots, O_t)$ contains a simple ant cycle. From Proposition 2 a_i must be a_1 , in order for the cycle to be a face. At a_1 the ant turns to the right (otherwise a_0 would be repeated) thus the edges (O_{t-1}, a_1) and (a_1, O_{t+1}) are strictly inside the shaded zone.

This proves the assertion.

Since a_0 is prohibited (the path must be simple), the assertion says that the ant needs to visit a_1 (or a_{-1}) before exiting the shaded zone. Let us call e_0 the initial position of the ant ($e_0 = (a_0, b)$), and e_1 the ant's position at time $t + 1$ (see Figure 4.3(b)). The boundary cells defined with respect to e_1 are, at the left side, exactly $a_2, a_3, \dots$, the same that for e_0 . Thus we can apply the assertion to e_1 , and prove that the ant must visit a_2 before exiting the zone. Applying the argument recursively, we see that the ant will never exit the zone.

□

Applying recursively Proposition 3 to each edge of each simple ant-path, we obtain that in fact the simple ant-paths are restricted to the sub-graph shown in Figure 4.4(b). This subgraph is a fractal tree of faces, with degree $k - 1$. In other words, it is composed

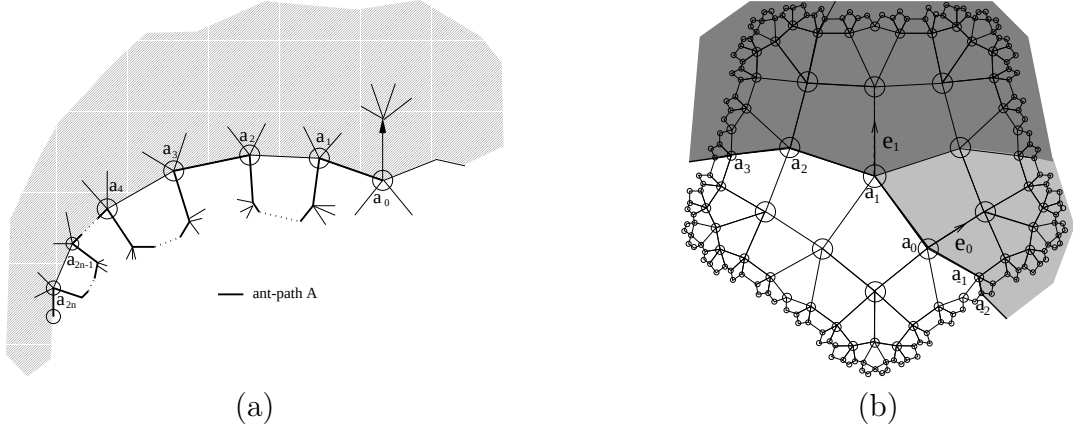


Figure 4.3: (a) The ant-path (A) going through the outside of the shaded zone and visiting all the boundary cells. (b) The limit cells to the left are the same for e_0 and e_1 .

by faces that, in the dual graph, correspond to a fractal tree of degree $k - 1$. It is a tree because, given two simple ant paths that bifurcate, we can apply Proposition 3 to the last common edge and to the two first different edges and we see that its respective zones are disjoint (see Figure 4.4(a)).

In order to depict the “tree”, let us remark that its branches are determined by the simple paths of the ant. For each simple infinite path, there is exactly one branch in the “tree”. It consists in the faces on which the ant turns (of which the ant goes along consecutive edges). For more precision, let us consider the following recursive definition.

Definition 4. *Given a graph satisfying (H) and an ant position (a_0, b) :*

1. *Let B be an empty set.*
2. *Color the face at the left of the ant position with white color, and the one at the right with black color. Add these faces to B .*
3. *For each face having only one adjacent face in B , add all of its adjacent faces except the one being to the right (if the face is white) of the adjacent face that is nearest of the ant. Color these new faces black. Make the analogous with the black faces.*

The subgraph composed by the cells and edges delimiting the faces of B is the tree of faces defined with respect to (a_0, b) .

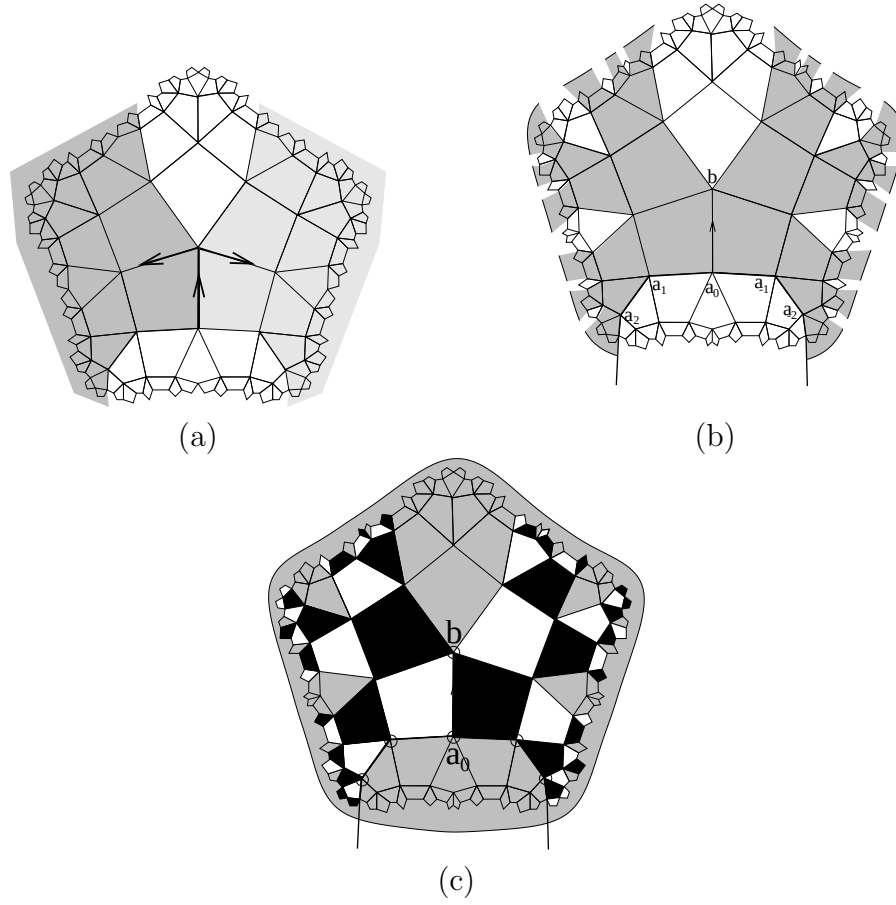


Figure 4.4: (a) Two paths that bifurcate will never join again. (b) Applying Proposition 3 to each reachable position of the ant, we see that the simple ant paths are restricted to a tree of faces. (c) We color the *tree* as a checkerboard.

This subgraph is defined independently of the degree of the graph. The cells have degree 4 or 5 in this subgraph, except a_0 , which has degree 3. Therefore, the result applies to all the graphs verifying (H), and we can enunciate the following proposition.

Proposition 4. *Let G be a graph verifying (H). Then no simple path can exit the subgraph depicted in Definition 4 when starting at the cells (a_0, b) .*

Let us introduce some notation that is needed for the proof of next proposition. If P , S and Q are paths, such that $P = (S, Q)$ is the concatenation of S and Q , then we say that $S = P \setminus Q$ and $Q = P \setminus S$. If the cells of the path Q belong to the path O and O is disjoint of S , then we also say that $S = P \setminus O$. We say that a path P contains an edge $\{u, v\}$ if the cells u and v appear consecutively in P .

Proposition 5. *Let G be a graph satisfying (H), c be a configuration and (v_0, v_1) be a position of the ant such that $c(v_0) = c(v_1)$. Let us consider the ant's trajectory $(v_n)_{n \in \mathbb{N}}$. Then for any $v_n \neq v_0$ there exists a simple ant-path that begins with the cells v_0 and v_1 and finishes with v_n .*

Proof. Let c be a configuration and (v_0, v_1) a position of the ant, the hypothesis says that $c(v_0) = c(v_1)$. Let us denote $P = (v_0, v_1, \dots, v_n)$ the path that the ant follows up to iteration n . To prove the proposition we must prove that there is a simple ant-path beginning at (v_0, v_1) and finishing at v_n .

The following algorithm defines a sequence of simple paths S_i . For any i , S_i is a simple path going from v_0 to v_i . They are composed by cells in P . We will prove that each one is an ant-path beginning at (v_0, v_1) or it is the single cell v_0 .

```

 $S_0 = (v_0)$ 
i=1
While(  $i \leq n$  ) do
  If (  $v_i \notin S_{i-1}$  )
     $S_i = (S_{i-1}, v_i)$ 
  Else
    Let  $j < i$  be the largest index such that  $v_i = v_j$ 
     $S_i = S_j$ 
  i = i + 1

```

The cell v_{i-1} belongs to S_{i-1} for any i , then the concatenation operation $S_i = (S_{i-1}, v_i)$ is well defined.

When the algorithm makes the assignation $S_i = S_j$ it makes in fact:

$$S_i = S_{i-1} \setminus (v_{j+1}, ..v_{i-1}) \quad (4.1)$$

Because, the cells in $S_{i-1} \setminus S_j$ belong to $(v_{j+1}, ..v_{i-1})$ and the cells $\{v_{j+1}, .., v_{i-1}\}$ does not belong to S_j .

The idea of the proof is to show that S_i is an ant-path for any i . This will be carried out by contradiction. If for some i , S_i is not an ant-path, we consider the smallest i such that this happens. The fact that S_h be an ant-path for any $h < i$ allows us to apply Proposition 4. We make a reconstructive work to infer the states of the concerned cells at this time and the path that the ant has followed. Through this, we prove that $i > m$ for m arbitrarily large. We conclude that i must be infinite. In the analysis we separate the case $v_i = v_0$. The condition that the states of v_0 and v_1 are initially equal is key to prove that the paths S_l begin with (v_0, v_1) for any l or it is a single cell.

To verify that a simple path is an ant-path, it is enough to regard the consecutive edges. They must belong to the same face. In general S_l is equal to $(v_0, v_{i_1}, v_{i_2}, .., v_{i_{k_l}})$, a path composed by cells in P . The sequence $(i_n)_{n=1}^{k_l}$ is increasing but it is not necessarily a sequence of consecutive numbers. The difference between the paths S_i and S_{i+1} , lies at their ends; S_{i+1} can have either one more cell or some less cells than S_i . So the length of S_i can increase of one unity or decrease of several unities when i is augmented of one unit. If it decreases, it does it of at least k unities, because k is the length of the shortest cycle.

If, in a given time, S_{i-1} is an ant-path and the cell $v_i \in S_{i-1}$, let $j < i$ be the largest index such that $v_j = v_i$; then the cycle $(S_{i-1} \setminus S_j, v_i)$ is a face and the situation is as Figure 4.5 shows. The cell v_{i+1} can either belong to S_{i-1} (the ant turns to the right in the figure) or not, depending on the state of cell v_i at time $i - 1$, which will be the opposite to the one at the last time the ant visited it.

The last time that the ant visited v_i was in time $j - 1$. We have that $v_j \in S_l$ for any l between j and i , otherwise, it cannot belong to S_{i-1} . Also $v_{j+1} \in S_l$ for any l between $j + 1$ and $i - 1$, because the sole way to be extracted is that a cell is repeated before it. But if such a cell is before v_j , also v_j will be extracted, and it cannot be v_j itself. This implies that v_{j+1} is where Figure 4.5 indicates.

If $j = 0$, the cell $v_i (= v_j = v_0)$ is visited for the first time at time i . Its state is the same as for the initial configuration and it is equal to the state of $v_1 (= v_{j+1})$ at time



Figure 4.5: When the ant arrives again at v_i it can go to the left or to the right, depending on the current state of this cell. S_{i-1} is marked with a bold line. When $v_i = v_0$, S_{i-1} begins at v_i . When $v_i \neq v_0$, S_{i-1} begins at other cell, since we suppose that it is an ant-path, then it must be as it is shown here.

0. A problem occurs if the cycle $S_{i-1}v_i$ is clockwise (counterclockwise) and v_i is in the *to-left* (*to-right*) state. Let us denote this situation **Case a)**. With reference to Figure 4.5, if v_i is in *to-left* state, $S_{i+1} = (v_0, v_{i+1})$ does not begin at (v_0, v_1) .

If $j > 0$, the cell v_{j-1} can belong to S_j or not (v_{j-1} can be different or equal to v_{i-1}). Let us analyze these two cases:

Case b) $v_{j-1} \in S_j$: This implies that $v_{j-1} \neq v_{i-1}$ (because $\{v_{j+1}, \dots, v_{i-1}\}$ is disjoint from S_j). With reference to Figure 4.6: at time $j - 1$ cell v_j was in *to-left* state, then it was in *to-right* state at time $i - 1$ and $v_{i+1} = v_{j+1}$, $S_{i+1} = S_{j+1}$, so, S_{i+1} is an ant-path. When $v_i = v_0$, S_{i+1} contains only two cells: v_0 and v_1 .

Case c) $v_{j-1} \notin S_j$: This implies that $v_{j-1} = v_{i-1}$ and that v_j was already a repeated cell, $v_j \in S_{j-1}$, more precisely $S_j \subset S_{j-1}$. In the figure, the state of cell v_j at time $j - 1$ is *to-right*, so at time i the ant turns to the left and S_{i+1} is not an ant-path. When $v_i = v_0$, S_{i+1} is an ant-path but it begins at a different edge than S_{i-1} .

At the beginning S_0 is a cell, and so, it is an ant-path. S_l will conserve this property until case c) occurs. S_l begins at (v_0, v_1) or it is a single cell until case a) or c) occurs. So, let us suppose that case a) or c) occurs, and let us consider the first time that this happens.

Case c) occurs if v_i has been a repeated cell also at time $j - 1$. Because time $i - 1$ is the first time that case c) occurs, when v_i is visited at time $j - 1$, case b) occurs. Let $h < j$ be the largest index such that $v_h = v_j = v_i$. Then the history must be as follows:

- In some time h , the cell v_i ($= v_h$) is added to S_{h-1} to obtain S_h .

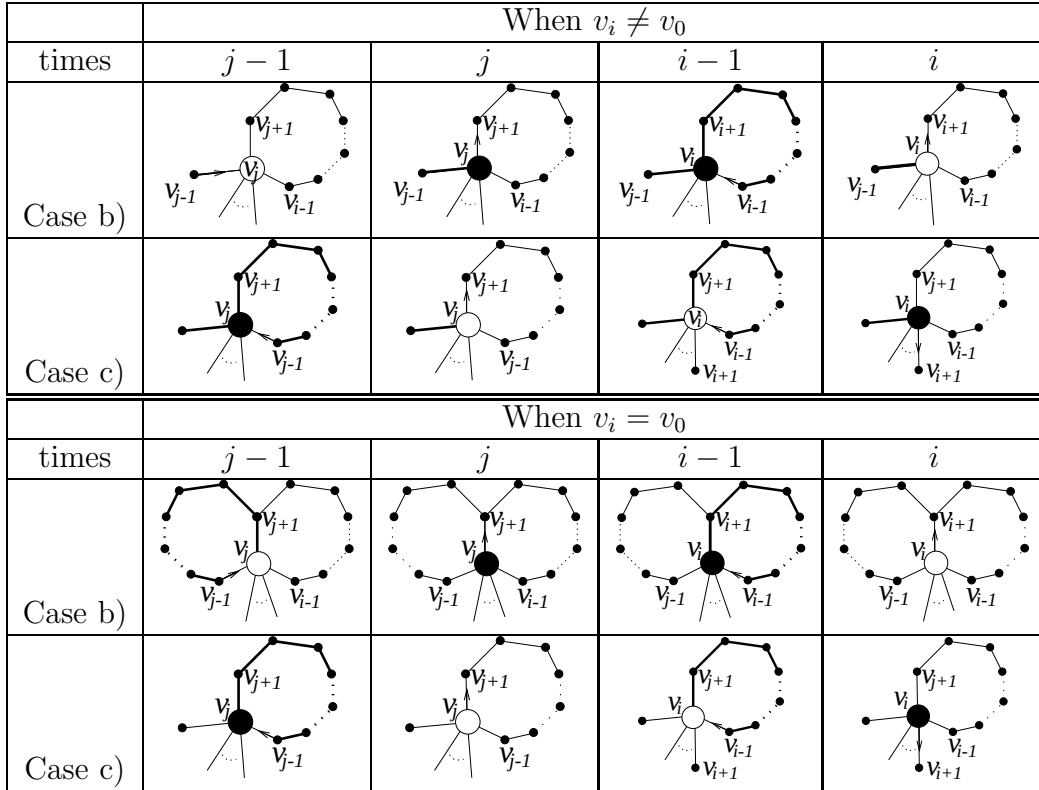


Figure 4.6: Cases b) and c) when $v_i \neq v_0$ and when $v_i = v_0$. The *to-left* state is represented with white color, and the *to-right* state with black color. The current path S_{j-1} , S_j , S_{i-1} or S_i is marked with bold line.

- v_h belongs to all the S_l for $l > h$ until it is visited at time $j - 1$ ($v_h = v_j$). Case b) occurs, so $v_{j+1} = v_{h+1}$.
- v_h belongs to all the S_l for $l \geq j$ until it is visited once more at time $i - 1$. Case c) occurs.

If case a) occurs, v_0 is not visited until time i and $v_1 \in S_l, (\forall l < i)$.

The paths $v_{h+1}, \dots, v_{j-1}$ and $v_{j+1}, \dots, v_{i-1}$ may not be simple, and we do not know their lengths. However, we know that they do not intersect $S_h (= S_j = S_i)$ and that $S_{j-1} \setminus S_j (= S_{i-1} \setminus S_i)$ is a simple ant-path that if it is joined with cell v_j forms a simple ant-cycle; it is a face as any other simple ant-cycle.

We emphasize that S_l is an ant-path for any $l \leq i$; this is the main fact of the proof. We can apply Proposition 4 to paths S_l for any $l \leq i$ starting from any edge in S_l .

It is true that

$$(v_{j+1}, v_{j+2}) \text{ belongs to } S_{i-1} \quad (4.2)$$

because to be erased, P must pass by a cell before v_{j+1} . If such a cell is v_{j+1} , Case b) arrives and the ant goes again to edge (v_{j+1}, v_{j+2}) , and we know that cells in S_j are not visited before time i .

For the following we suppose, without loss of generality, that S_{i-1} is as Figure 4.5 shows. It may be like this or a reflection of this. We divide our analysis into two parts: **i)** We analyze first the states and the trajectory of the ant between times h and j . **ii)** With this information we prove an assertion that says that the ant's trajectory between times h and j , and j and i is infinite.

i) About the ant's trajectory before time j and the states of the cells at this time.

From equation 4.2 we conclude that the ant turns to the right at time $j + 1$, which implies that:

$$\text{The state of cell } v_{j+1} \text{ at time } j \text{ is } \textit{to-right} \quad (4.3)$$

The first consequence of this is that case a) cannot occur, because the cycle (S_i, v_i) is clockwise and from the hypothesis of the proposition, v_0 ($j = 0$) is in the same state that v_1 (v_{j+1}), so it is in the *to-right* state.

Now we know that case c) occurs at time $i - 1$.

Assertion 1: *Given that, for some $m < i$, (S_{m-1}, v_m) contains a clockwise (counter-clockwise) oriented cycle $(a_1, \dots, a_k)$, then:*

1) *If a cell a_l (with $l > 1$) of the cycle is in to-right (to-left) state at time m , and we denote $t < m$ the largest index such that $v_t = a_l$, then (S_{t-1}, v_t) contains the cycle that is to the left (to the right) of the arrow that goes from a_l to a_{l+1} .*

2) *a_k is necessarily in to-left (to-right) state at time m .*

1) The last time that the ant visited a_l (at time $t - 1$) it exits it by the edge $\{a_l, a_{l+1}\}$, otherwise $\{a_l, a_{l+1}\}$ cannot belong to S_{m-1} . The cell a_l ($= v_t$) is in the *to-left* state at time $t - 1$. With reference to Figure 4.7, this means that it comes from cell c (i. e., $v_{t-1} = c$). But c is not in S_{m-1} , then, there are two possibilities: i) it is erased after t , ii) it is erased exactly in time t . i) is not possible because if c is erased also a_l does. So, $a_l \in S_{t-1}$, a cycle is formed and case b) arrives. Then $\{a_l, a_{l+1}\}$ belongs to this cycle. The unique cycle containing both edges $\{a_l, a_{l+1}\}$ and $\{c, a_l\}$ is the cycle to the left of the pair (a_l, a_{l+1}) . Which proves 1).

2) If at time m , a_k is in *to-right* state, we know by 1) that the ant travels through the cycle to the left of the pair (a_k, a_1) . But this means that case c) occurred at cell $a_1 = v_m$, and contradicts the hypothesis.

Q.E.D.

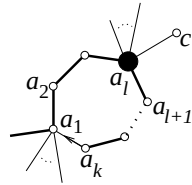


Figure 4.7: If a_l is in *to-right* state, the ant came from cell c the last time that it visited it.

This assertion gives us information about the ant's trajectory in function of the current states of the cells in the cycles that it has traveled through. For each cycle that the ant has traveled through and that has some cells in certain state, we know that the ant has

traveled through some others adjacent cycles and so on. Since we are supposing that the ant comes back to v_i at time j , there must exist some cycle that the ant traveled through directly, and such that all its cells are in the *to-left* state if the cycle has been traveled through clockwise or in the *to-right* state if it has been traveled through counterclockwise.

The hypothesis of the assertion is that in some time the path (S_{m-1}, v_m) contains a cycle. But we can see that for any cell visited by the ant there is an time such that it belongs to a cycle of the form (S_{m-1}, v_m) . In fact, the simple ant-path that joins v_0 with any cell is unique (from Proposition 4). The last time that the ant visits a given cell v_t , S_t is a simple ant-path that goes from v_0 to v_t . If v_t does not belong to $S_j - 1$, it must be extracted in some time before $j - 1$, i. e., there exists an time m larger than t and less than j such that S_{m-1} contains v_t and S_m does not contain it. Which implies that (S_{m-1}, v_m) contains a cycle. Since we are considering the last time that the cell is visited, its state at this time will remain the same at time j . So we can use Assertion 1 to study the ant's trajectory after j .

ii) Now we have the tools to examine the ant's trajectory between time j and i :

We will prove Assertion 2. It talks about the consequences of Assertion 1 on the trajectory of the ant between j and i . It says that the ant will follow a simple path composed by cells that it traveled through between time h and j .

Assertion 2: *After time j , the ant follows a simple path beginning at v_i that turns at least one time (and at most $k - 2$ times) to the right and after it turns alternately d_m times to the left and to the right, with $2 \leq d_m \leq k - 2$, for all $m \in \mathbb{N}$. Moreover, the faces on which the ant turns were already traveled through by the ant between time h and j .*

We will prove this by induction in the number, m , of changes of turn direction. For $(m = 0)$: from Assertion 1, the last cell of the cycle to the right of (v_i, v_{j+1}) is in *to-left* state at time j . The cell v_{j+1} is in *to-right* state as we established above. Then the ant will turn to the right at least one time and at most $k - 2$ times before to find a cell in *to-left* state. Let us call this cell and the former one, b_0 and a_0 respectively; b_0 is not v_i . Also we verify that the cycle that is to the left of the arrow (a_0, b_0) was already visited by the ant between times h and j (see Figure 4.8(a)).

For $(m + 1)$: let us suppose that the ant turns alternately d_l times to the left and to the right, with $2 \leq d_l \leq k - 2$, for all $l \leq m$. Let us call (a_l, b_l) the positions of the ant before changing of turn direction. From the induction hypothesis, the ant turns to the left in b_l if l is even and to the right if l is odd.

Let us consider without loss of generality, that m is even. Also we know by induction hypothesis that the ant traveled through the cycle that is to the left of (a_m, b_m) . Then we can apply Assertion 1 to this cycle: it says that the cell in the cycle that is to the left of a_m is in *to-right* state (see Figure 4.8(a), cell α). Since b_m is in *to-left* state, the ant enters the cycle. It will turn to the left d_{m+1} times before finding a cell in *to-right* state. Assertion 1 allows us to say that $d_{m+1} \leq k - 2$.

If the first cell that is in *to-right* state is the cell to the left of b_m , let us call this cell c_m , we can apply this proposition with respect to position (b_m, c_m) , because b_m and c_m are in the same state. Then, the cells that the ant reaches before time i can be also reached through a simple path that begins at (b_m, c_m) . Then we apply the Proposition 3 and we see that v_i is out of the shaded zone of cells that the ant can reach through simple paths, which is a contradiction. To see this more clear, let us notice that the ant will never visit a cell in the cycle to the right of (a_m, b_m) with exception of a_m and b_m as Figure 4.8 shows. From the figure we can verify that any simple path joining c_m with v_i that does not visit any other cell in the cycle, joined with S_m contains a cycle involving more than one face, then there is no simple path from b_m to v_i . We conclude that c_m is in the *to-left* state and that $d_{m+1} \geq 2$.

That means that in the cycle there is a cell in the *to-left* state followed by a cell in the *to-right* state, say (a_{m+1}, b_{m+1}) , Assertion 1 says that the cycle to the right of this pair of cells was already visited by the ant. With this last fact we prove the assertion for $m + 1$.

Assertion 2 shows that after time j the ant follows a simple path that oscillates constantly. Such a path cannot visit v_i because this would imply the existence of a cycle different from a face. We conclude that case c) never arrives and S_n is a simple ant-path joining v_0 and v_n . In that way we have proved that any cell being reachable by the ant, is also reachable through a simple ant-path. $\square$

Theorem 5. *In a graph satisfying (H), if in some time t the ant is between two cells (a, b) in the same state, then its trajectory after time t does not exits from the subgraph defined by recurrence in Definition 4.*

The ant is between two cells in the same state with a frequency larger than $1/k$. In other words, it cannot avoid this situation for more than k steps, for k consecutive equal states would bring it back to the first cell. It follows that the ant will be always restricted to a subgraph like the one described above.

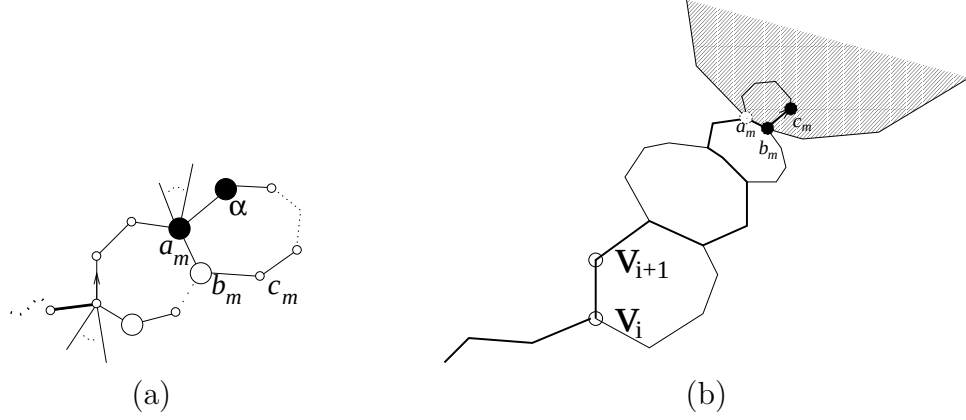


Figure 4.8: (a) The ant necessarily goes into the cycle to the left of cells (a_0, b_0) . Then, from Assertion 1, the cell α is in the *to-right* state. (b) The shaded zone describes the cells that are reachable by the ant through simple paths beginning at (b_m, c_m) .

This result is strong and it carries on many consequences. First, let us remark that the *tree* does not change with the degree of the underlying graph. *The dynamics of the ant will be the same for all the $\Gamma(k, d)$ graphs with a fixed k satisfying (H).* The unboundness of the ant's trajectory on the even degree graphs with $k \geq 4$ is, then, inherited for all the $\Gamma(k, d)$ graphs with $k \geq 4$ and $d \geq 5$. It can be also extended to the $\Gamma(3, d)$ graphs because any finite subgraph contained in the tree of faces has a corner. Also the property that the ant passes over each edge always in the same direction is inherited.

If we want to predict the trajectory of the ant, it is enough to look at the cells on the *tree*. This will allow us to prove that the ant's trajectory is eventually regular when starting over a finite initial configuration.

Since the ant is its own inverse, the past trajectory of the ant is also restricted to a similar *tree* but in the inverse position. In this way we know the cells that may belong both to the past and to the future of the ant. Those cells constitute the *memory of the ant*. Let us take the case $k = 3$ as example. In this case, the *tree* has degree 2, it is a simple chain of tiles (see Figure 4.9). This fact was already proved in [25]. The *memory of the ant* is on the triangles that are at the two sides of the ant. In this particular case the ant's memory is finite. The consequences of this will be analyzed in Chapter 5 together with other properties that follow from Theorem 5.

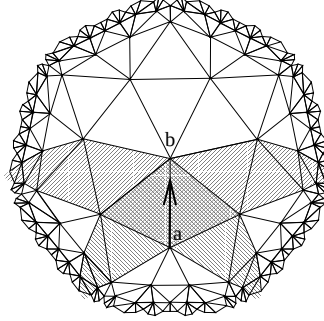


Figure 4.9: The intersection between the future reachable cells and the past possible reached cells are the two triangles around the ant. We call them the *memory of the ant*.

4.2 Eventually regular trajectories

The open question in the grid has a positive answer on the graphs verifying (H). For any finite initial configuration the ant eventually falls in a regular trajectory, the eventual motion is always the same and his period is $(k - 1)(k + 1)$.

Theorem 6. *Let G be a graph verifying (H). Starting over any finite initial configuration, the ant's trajectory is eventually regular and the period of the eventual behavior is always $(k - 1)(k + 1)$.*

Proof. The idea of the proof is the following:

Without loss of generality, let us suppose that there is a finite number of cells in *to-right* state. Considering (at each branch of the tree) the farthest face containing a node in *to-right* state, we observe that there are only 2 ways for the ant to arrive to it. Analyzing each case, we observe that in one of them, the ant arrives to be between two nodes in the same state and the accessible zone, from its position, does not contain the other cells in *to-right* state. The configuration of the accessible zone is always the same, let us call it $\mathcal{C}$.

In the second case, the ant rebounds, but this branch is left in a configuration such that the next time the ant enters it, it will find the first case.

Simulating the ant on $\mathcal{C}$, we verify that its trajectory is regular, with period $(k - 1)(k + 1)$.

Now, let us develop the proof.

Given a finite initial configuration and an initial ant position, let us consider a time t such that the ant is between two cells in the same state. So, we can apply Theorem 5 to assure that the ant's trajectory is restricted to a tree of faces.

Without loss of generality, we suppose that there is a finite number of cells in *to-right* state. We consider the farther cells containing cells in *to-right* state. These cells are the boundary of a zone that contains all the cells in *to-right* state, which we call *initial support*. Each face can be entered for the first time by the ant only by one of its edges, the one that is nearest to the ant initial position.

Since the ant's trajectory is unbounded, there exists a time t' such that the ant exit one of these faces to a face whose cells are all in *to-left* state, except for the cells on the input edge. Depending on the states of these two cells, two possibilities appears, see Figure 4.10.

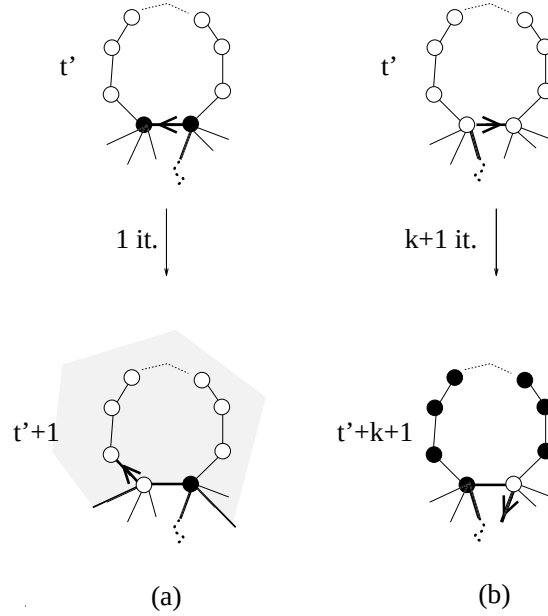


Figure 4.10: The ant can enter a face in two different ways. In case (a) the ant future trajectory is restricted to the shaded zone, and it does not includes any of the cells in the past trajectory. In case (b) the ant makes a tour and comes back.

In case (a), at time $t' + 1$ the ant is between two cells in the same state and the shaded zone with respect to its position does contains only one cell in *to-right* state. In case (b), the ant travel through the face, and at time $t' + k + 1$ the ant exit the face coming back to the initial support. The cells in the face are now in *to-right* state.

But the ant will exit the initial support ones more. Case (b) may arrive many more times. But in some time, case (a) arrives or the ant enter a face outside the initial support that has been already visited and whose cells are in *to-right* state, as it left it after case (b). Figure 4.11 shows a simulation of this situation, case (a) arrives on an adjacent face.

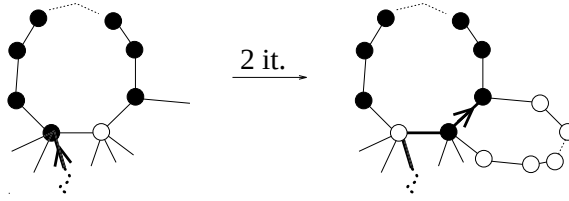


Figure 4.11: When the ant comes back to a face that it visited in case (b), case (a) arrives in a neighbor face.

We have proved that for any finite initial configuration the ant eventually arrives to a position such that all its reachable cells are in *to-left* state, except for one cell, as it is shown in Figure 4.12(a). At this point, we need only to let evolves the ant on this configuration to verify the regularity of the ant's trajectory; the same configuration appears periodically. Figure 4.12(b) shows this. $\square$

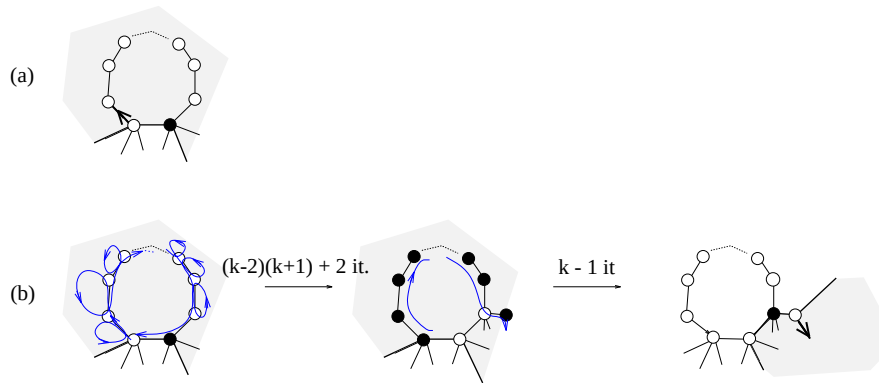


Figure 4.12: (a) The ant always arrives to this configuration. (b) The ant's evolution on (a), after $(k - 1)(k + 1)$ iterations the ant arrives to (a) again.

Chapter 5

Symbolic Dynamics

In this chapter we model the ant as a topological dynamical system. A topological dynamical system is a topological space X called “phase space”, together with a continuous function $f : X \rightarrow X$.

There are many ways to formalize the ant as a dynamical system, depending on the choice of the representation of the system state. This problematic is discussed in [28] for the case of Turing machines and different properties are found for different representations of the system. In [5] a system similar to the ant is modeled as topological dynamical system.

In Definition 1, the system state is described as a configuration $c : V \rightarrow \{to-left, to-right\}$, together with the ant position. Then, the set of states is $A_G = \{to-left, to-right\}^V \times O$. Where O represents the set of pairs $(i, j) \in V \times V$, such that $\{i, j\} \in E$.

The transition function acts over $c(j)$ and (i, j) . With the product topology, A_G is a closed non-compact set. The transition function, say $f : A_G \rightarrow A_G$, is continuous and one-to-one.

This representation is the more natural one, and it is the best for simulation. Nevertheless, it is not adequate for the analysis of the ant’s movement. The state representation contains too much information; in fact, the cells that are never visited by the ant, are not interesting. Shifting a configuration, together with the ant position does not change the ant’s movement.

To emphasize the regularities of the ant’s movement, regardless of the changes in the

configuration, we define another dynamical system that is related with (A_G, f) through a reduction function as follows.

Let $\pi : A_G \rightarrow \{L, R\}$ be the function defined by:

$$\pi(c, (i, j)) = \begin{cases} L & \text{if } c(j) = \text{to-left} \\ R & \text{if } c(j) = \text{to-right} \end{cases}$$

The sequence $(\pi(f^n(x)))_{n \in \mathbb{N}}$ contains the states of all the cells that the ant will visit when the starting state is x . At the same time, it is itself a precise description of the ant's trajectory. If we know the starting ant's position and $(\pi(f^n(x)))_{n \in \mathbb{N}}$, it is possible to infer the whole ant's trajectory.

Defining the set:

$$H_G = \{(\pi(f^n(x)))_{n \in \mathbb{N}} \in \{L, R\}^{\mathbb{N}} \mid x \in A_G\}$$

and the shift function $\sigma : H_G \rightarrow H_G$ by $\sigma((a_0, a_1, a_2, \dots)) = (a_1, a_2, a_3, \dots)$, we have the following scheme:

$$\begin{array}{ccc} A_G & \xrightarrow{f} & A_G \\ \psi \downarrow & & \downarrow \psi \\ H_G & \xrightarrow{\sigma} & H_G \end{array}$$

With $\psi(x) = (\pi(f^n(x)))_{n \in \mathbb{N}}$.

We shall prove that H_G is a closed set and that ψ is onto and continuous.

In spite of ψ not being one-to-one, $\psi(x)$ contains all the information that is relevant in the ant's dynamics. In some sense, all the points in $\psi^{-1}(\psi(x))$ are equivalent. For that we think that the properties of (H_G, σ) reflect the complexity of (A_G, f) even if they do not coincide, in general, with the (A_G, f) 's properties. On the other hand, (H_G, σ) is a simple and homogeneous representation of (A_G, f) , in the sense that $(H_G, \sigma) \subset \{L, R\}^{\mathbb{N}}$ for any G , then we work within a single space, we apply the same tools and describe the different properties in the same language for any graph.

In the next section we give some of the basic definitions on dynamical systems and symbolic dynamics. Based on the first chapters we prove several properties of the system (H_G, σ) which are listed and proved in Section 5.2. Finally, Sections 5.3 and 5.4 establishes a relation between dynamical properties of (H_G, σ) and computational complexity of the ant. This is achieved by modeling a Turing machine as a system on the style of (H_G, σ) .

5.1 Definitions on symbolic dynamics

For the following we suppose that we have a dynamical system (X, f) where X is a compact metric space.

A point u is said to be **periodic** if there exists a number $k \in \mathbb{N}$ such that $f^k(u) = u$, it is said to be **preperiodic** if there exists a number $t \in \mathbb{N}$ such that $f^t(u)$ is periodic. A point u is said to be **sensitive to initial conditions** if for any positive numbers δ and ϵ there exists a second point v , such that $d(u, v) < \epsilon$ and $d(f^n(u), f^n(v)) > \delta$ for some integer n .

A system (X, f) is said to be **transitive** if and only if $(\forall U, V \subset X, U, V \text{ open})(\exists n \in \mathbb{N}) f^n(U) \cap V \neq \emptyset$. It is said to be **mixing** if and only if $(\forall U, V \subset X, U, V \text{ open})(\exists n_0 \in \mathbb{N})(\forall n \geq n_0) f^n(U) \cap V \neq \emptyset$. It is said to be **positively expansive** if and only if $(\exists \epsilon > 0)(\forall u \neq v)(\exists n \geq 0) d(f^n(u), f^n(v)) > \epsilon$.

In a system that is positively expansive, all the points are sensitive to initial conditions.

Definition 5. A set $Y \subset X$ is **f-invariant** (resp. strictly f -invariant) if and only if $f(Y) \subset Y$ (resp. $f(Y) = Y$). If Y is strictly invariant and topologically closed, Y is compact and the pair (Y, f) is a **subsystem** of (X, f) .

We work particularly with symbolic systems. In the following we define it and we recall some of the basic definitions of the area.

A *symbolic system* is defined as follows: let A be a finite set that we will call *alphabet*. $A^* = \cup_{n \in \mathbb{N}} A^n$ is the *set of words on A* and $A^{\mathbb{N}} = \{u = (u_i)_{i \in \mathbb{N}} | (\forall i \in \mathbb{N}) u_i \in A\}$ is the set of infinite sequences of elements of A . The following metric makes $A^{\mathbb{N}}$ compact:

$$d(u, v) = \begin{cases} 0 & \text{if } u = v \\ \frac{1}{2^{N(u, v)}} & \text{if } u \neq v \end{cases}$$

where $N(u, v) = \min\{n \in \mathbb{N} \mid u_n \neq v_n\}$.

The sphere of radius $1/2^n$ and center $a = (a_i)_{i \in \mathbb{N}}$ is the set $[a_0 a_1 \dots a_n] = \{u \in A^{\mathbb{N}} \mid \forall i \leq n, u_i = a_i\}$. The spheres in this metric space are commonly called “cylinders”.

In $A^{\mathbb{N}}$ we consider the shift function $\sigma : A^{\mathbb{N}} \rightarrow A^{\mathbb{N}}$ defined as follows:

$$\text{for } u = (u_n)_{n \in \mathbb{N}} \in A^{\mathbb{N}}, \quad \sigma(u)_i = u_{i+1} \quad (\forall i \in \mathbb{N})$$

The shift function σ is continuous and onto. Thus, the pair $(A^{\mathbb{N}}, \sigma)$ is a topological dynamical system. It is called **$|A|$ -full shift** and it is expansive and mixing.

A subsystem of a full shift is called **subshift**.

Given a subshift $Y \subset A^{\mathbb{N}}$ the *language associated to Y* is defined as the set:

$$L(Y) = \{a_1 \dots a_n \in A^* \mid (\exists u \in Y) u_1 \dots u_n = a_1 \dots a_n\}$$

It is composed by the finite subsequences of elements of Y . The language of a subshift defines it completely, i. e., only Y has $L(Y)$ as associated language. This means that to verify if a sequence belongs to Y it is enough to verify if all its finite subsequences belong to $L(Y)$.

Three important classes of subshifts are: the subshifts of finite type, the sofic systems and the coded systems.

Definition 6. A **labeled graph** is a pair $(G = (Q, E), \lambda)$, where G is a directed graph and λ is a function whose domain is E and its image is a finite set called the set of “labels”.

A labeled graph is a **Fischer automaton** if Q is countable and for any label l , and any vertex q there exists at most one vertex q' such that $(q, q') \in E$ and $\lambda(q, q') = l$.

Given a labeled graph $\mathcal{G} = (G = (V, E), L)$ we define the language recognized by it, as:

$$\mathcal{L}_{\mathcal{G}} = \{u \in A^* \mid \text{There is a path } (a_i)_{i=0}^n \text{ of } G \text{ such that } (\forall i \in \{0, \dots, n\}) u_i = \lambda(a_i)\}$$

A language is **coded** if it is recognized by Fischer automaton with G strongly connected. A language is **regular** if a finite Fischer automaton recognizes it.

In the same way we say that a subshift Y is **sofic** (resp. **coded**) if the associated language $L(Y)$ is regular (resp. coded).

If the language of a sofic subshift Y is recognized by a Fischer automaton satisfying that there exists an integer n such that for any word w of length n in $L(Y)$ there exists a unique sequence $(a_i)_{i=1}^n$ such that $(\lambda(a_i))_{i=1}^n = w$, then Y is called **subshift of finite type**.

Let us notice that for any subshift there exists a countable Fischer automaton $\mathcal{G}$ that recognizes it, but it is not true that for any transitive subshift there exists a countable and strongly connected Fischer automaton $\mathcal{G}$ that recognizes it. The strong connexity of G gives to $(\mathcal{L}_{\mathcal{G}}, \sigma)$ a stronger property than only transitivity.

It is easy to see that a sofic and transitive subshift is coded and that any subshift of finite type is sofic.

More details about *dynamical systems* and *subshifts* can be found in [13, 26, 45].

5.2 Ant's properties

We have studied the properties of $(H_{\Gamma(k,d)}, \sigma)$ and their dependence on k and d . They are listed in the following table, we assume that k and d are such that $\Gamma(k, d)$ is an infinite graph.

Transitive	Mixing	Positive Entropy	Coded System		not Sofic	SFT
$d \geq 4$	$d \geq 4$ $k \geq 4$	always	$d \geq 5$ $k \geq 4$	$d \geq 6$ $k \geq 3$	$k \geq 4$	$k = 3$

A subshift is always positively expansive so this property is true for any k and d . Next we prove all of these properties.

First we must prove that the function ψ is continuous and that H_G is closed and invariant, in order to establish that (H_G, σ) is a subshift:

Let $(c_0, (i_0, j_0)) \in A_G$ and $(b_t)_{t \in \mathbb{N}} = \psi((c_0, (i_0, j_0)))$. Let $\epsilon > 0$ and $n = \lceil \log_2(1/\epsilon) \rceil$. The sphere with center b and radius ϵ is the cylinder $[b_0 \dots b_n]$. Let us define $(c_t, (i_t, j_t)) = f^t(c_0, (i_0, j_0))$, for $0 < t \leq n$, in which $(j_t)_{t=0}^n$ are the cells that the ant visits between iteration 0 and n . Let us consider the set

$$\Theta = \{d \in \{to-left, to-right\}^V \mid (\forall i \in \{0, \dots, n\}) d(j_t) = c_0(j_t)\} \times \{(i_0, j_0)\}$$

It is direct that $(c_0, (i_0, j_0)) \in \Theta$ and that Θ is an open set. It is clear that any configuration that coincides with c_0 in $(j_t)_{t=0}^n$ makes the ant to follow the same trajectory as c_0 , at least in the first n steps. So, $\psi(\Theta) \subset [b_0 \dots b_n]$. Then ψ is a continuous function.

H_G is *shift-invariant*, because applying the shift to a sequence is equivalent to apply the ant's rule to the corresponding configuration: $\sigma(H_G) = \sigma(\psi(A_G)) = \psi(f(A_G)) = \psi(A_G) = H_G$. It is a closed space, because if we have a convergent sequence in H_G , $(a^t)_{t \in \mathbb{N}} \rightarrow a$, there exists a creasing sequence of integers $\{t_n\}_{n \in \mathbb{N}}$ such that $a^m \in [a_0 \dots a_n]$, $\forall m \geq t_n$. The sequence a^m coincides with a in the first n letters. From continuity of ψ we know that the set $\psi^{-1}([a_0 \dots a_n])$ is closed. Let us consider the set $I_e = \{to-left, to-right\}^V \times \{e\}$, the set of states such that the ant's position is e . I_e is closed and complete, because it is the product of complete sets. Then,

$$\bigcap_{n \in \mathbb{N}} \psi^{-1}([a_0 \dots a_n]) \cap I_e \neq \emptyset$$

and

$$\emptyset \neq \psi\left(\bigcap_{n \in \mathbb{N}} \psi^{-1}([a_0 \dots a_n])\right) \subset \bigcap_{n \in \mathbb{N}} [a_0 \dots a_n] = \{a\}$$

which implies $a \in \psi(A_G)$, i. e., H_G is closed.

The language associated to H_G is the set of words describing an ant-path. In this context, we will identify the notion of ant-path with the notion of word, because both refer to the same object.

Some previous definitions and properties of bi-regular graphs are needed.

Definition 7. Let us say that a set $C \subset V$ is **convex** if any cell $u \notin C$ satisfies that for any edge $\mathbf{e}$ between $\mathbf{u}$ and a cell in C , there exists an infinite zigzagging path starting from $\mathbf{e}$ that does not intersect C .

Given a set of cells s , we define the **convex cover** of s , $ev(s)$ as the smallest convex set

that contains s .

A **sphere**, $B(s, R)$, of radius R and center s is the set of cells whose distance to s is less or equal than R .

A **geodesic** is a path joining two cells and having minimal length.

Proposition 6. *In the infinite $\Gamma(k, d)$ graphs, the spheres are convex.*

Proof. Let us consider a cell a outside the sphere, whose neighborhood intersects the sphere: $N(a) \cap B(s, R) = \{u_0, \dots, u_n\}$. The cells $\{u_0, \dots, u_n\}$ must be at distance R from s , otherwise a belongs to $B(s, R)$. This implies that for each cell u_i with i from 0 to n there exist geodesics that join it to s . So, there are $n + 1$ geodesics from a to s . In [37] it is proved that for hyperbolic graphs, the set of geodesics between two points have no interior points. This implies that in hyperbolic graphs n must be less or equal than 1.

For Euclidean graphs (the case of the triangular, hexagonal and square grids) n must also be ≤ 1 but the arguments are different, essentially is due to the existence of a coordinate system.

This implies that the edges $\{a, u_0\}$ and $\{a, u_1\}$ belong to the same face. So for any of them, one of the neighbor edges is not adjacent to the sphere; we can define a zigzagging path that begins at one of these edges continues on a neighbor edge and that goes in the direction opposed to the sphere. $\square$

If s is finite, $ev(s)$ is also finite, because $ev(s)$ is contained in some sphere of finite radius.

Proposition 7. *$(H_{\Gamma(k, d)}, \sigma)$ is transitive if $d \geq 4$.*

Proof. It is enough to prove it just for the open sets of the neighborhood base, i.e., the cylinders. Let $[u_0 \dots u_r]$ and $[v_0 \dots v_s]$ be two non-empty cylinders in $H_{\Gamma(k, d)}$. We must prove that $(\exists n \geq 0) P_n = [u_0 \dots u_r] \cap \sigma^{-n}([v_0 \dots v_s]) \neq \emptyset$.

P_n is the set of sequences in $H_{\Gamma(k, d)}$ that begin with $u_0 \dots u_r$ and after n steps it contains the factor $v_0 \dots v_s$.

Then we must find an initial configuration such that the ant begins with $u_0 \dots u_r$, and after a delay it makes $v_0 \dots v_s$.

By continuity of ψ , there exist two open sets O_u, O_v in A_G , such that $\psi(O_u) = [u_0 \dots u_r]$ and $\psi(O_v) = [v_0 \dots v_s]$.

We can choose two configurations x_u and x_v in O_u and O_v , respectively. Let us define s_u the cells in V that are visited by the ant before iteration r , when starting with configuration x_u . Analogously, define s_v . They are finite sets.

In Chapter 4 we have seen that the ant's trajectory is not bounded for any initial configuration when the graph is a $\Gamma(k, d)$ graph and d is odd or $d \geq 5$, provided that the graph is infinite. Thus we know that, starting from x_u , the ant will exit from $ev(s_u)$ in a given instant t_u (with $t_u > r$). In the same way, if we consider the evolution of the ant rule in reverse time, the trajectory is also unbounded, and we know that, starting from x_v , the ant will exit from $ev(s_v)$ in an instant $-t_v$ (with $t_v > 0$).

The sequence x_u can be modified on the cells that are not in s_u , in such a way, we do not affect the fact that $\psi(x_u) \in [u_0 \dots u_r]$. The idea of the proof consists in changing the states of the cells outside of $ev(s_u)$ so as to make the ant to follow an arbitrary long path after time t_u . And to "paste" the restriction of x_v to $ev(s_v)$ in some part of V disjoint of $ev(s_u)$ such that the ant arrives to this part to finally follow the sequence $v_0 \dots v_s$.

Let $(d, (e_1, e_2)) = f^{t_u}(x_u)$ the configuration at time t_u . The cell e_1 belongs to $ev(s_u)$ and e_2 does not. Then, by definition, there exists an arbitrary long zigzagging path that begins on (e_1, e_2) and does not intersect $ev(s_u)$.

Let $(b, (g_1, g_2)) = f^{-t_v}(x_v)$. The cell g_2 belongs to $ev(s_v)$ and g_1 does not. For the same reason, there exist a zigzagging path that begins at (g_2, g_1) and does not intersect $ev(s_v)$.

We can shift and rotate b so as to align the two zigzagging paths in such a way that the zigzagging path of s_u finishes at s_v and that $ev(s_u)$ does not intersect $ev(s_v)$. Let m be the length of the zigzagging path.

So we define a configuration that coincides with x_u on $ev(s_u)$, and with $f^{-t_v}(x_v)$ on $ev(s_v)$. We define the states of the cells on the zigzagging path to make the ant to follow it from $ev(s_u)$ to $ev(s_v)$. The configuration, say x , defined in that way satisfies that $\psi(x) \in P_n = [u_0 \dots u_r] \cap \sigma^{-n}([v_0 \dots v_s])$, with $n = (t_u - r) + m + t_v$. $\square$

If $d = 3$, there exist bounded ant's trajectories, a cylinder $[w_0 \dots w_s]$ where w is a path that contains a complete cycle of a bounded ant's trajectory contains only one element! It is easy to find another word v such that $[v_0 \dots v_r] \cap \sigma^{-n}([w_0 \dots w_s]) = \phi$.

Proposition 8. $(H_{\Gamma(k, d)}, \sigma)$ is **mixing** if $d \geq 4$ and $k \geq 4$.

Proof. We must prove that for any pair of non-empty cylinders $[u_0 \dots u_r]$ and $[v_0 \dots v_s]$ there

exists a number n_0 such that for any number $n \geq n_0$ the set $P_n = [u_0 \dots u_r] \cap \sigma^{-n}([v_0 \dots v_s])$ is not empty.

The proof is the same as the one of previous proposition, with the difference that here we must verify that the length of the zigzagging path is arbitrary.

First we remark that there is a parity restriction on this length that depends on the direction of the first and the last turns of the zigzagging path. They may be fixed by the shape of $ev(s_u)$ and $ev(s_v)$. Figure 5.1 shows an example of this situation in the case of the square grid.

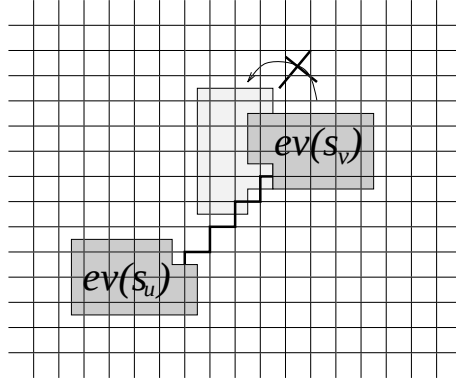


Figure 5.1: $ev(s_v)$ cannot be rotated; a shift can be only of even length.

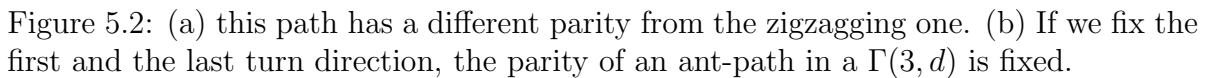
When $k \geq 4$ this situation can be avoided by choosing the path between $ev(s_u)$ and $ev(s_v)$ other than a zigzagging path (see Figure 5.2(a)), which proves the proposition.

When $k = 3$ there is no other simple path than the zigzagging one. Ant-paths contains necessarily an even number of cycles so there is no any odd delay (see Figure 5.2(b)). $(H_{\Gamma(3,d)}, \sigma)$ is not mixing. $\square$

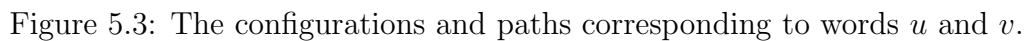
Proposition 9. $(H_{\Gamma(k,d)}, \sigma)$ has a positive topological entropy.

Proof. The idea of the proof is to see that there exist two words that can be concatenated arbitrarily and the resulting word is always an ant-path. This implies that the number of words in $L(H_{\Gamma(k,d)})$ grows exponentially with the length word which means that the entropy of $H_{\Gamma(k,d)}$ is positive. These words are $u = (RL)^{k+2}$ and $v = RLR^k L^k RL$.

It is easy to verify that u and v belongs to $L(H_{\Gamma(k,d)})$. Figure 5.3 shows the paths that they describe. Both are contained in a zigzagging path. The concatenation of the



In the graphs $\Gamma(k, d)$ a zigzagging path never intersects itself, then we can concatenate an infinite number of copies of these configurations.



The topological entropy is defined by,

Where $\sharp(m, H_{\Gamma(k,d)})$ is the number of words of length m in $H_{\Gamma(k,d)}$.

80

$$\begin{aligned}
h_t(H_{\Gamma(k,d)}, \sigma) &\geq \lim_m \frac{1}{n(2k+4)} \log(\sharp(m, H_{\Gamma(k,d)})) \\
&\geq \frac{1}{2k+4} \log 2 > 0
\end{aligned}$$

□

Chaoticity of a dynamical system has different definitions depending on the context [13]. Some people say that a system is chaotic if it is expansive and transitive; others ask only every point to be sensitive to initial conditions. To have positive entropy is also a characteristic that talks about chaoticity. All this criteria are satisfied by (H_G, σ) , so we can say that it is a chaotic system.

Now we will study the properties associated to the language of $H_{\Gamma(k,d)}$. These properties can be interpreted in (A_G, f) as the correlation of the future ant's trajectory with its past trajectory.

Proposition 10. $(H_{\Gamma(k,d)}, \sigma)$ is **not a sofic system** when $k \geq 4$.

Proof. Let us suppose that $(H_{\Gamma(k,d)}, \sigma)$ is sofic. Then, there is a Fischer automaton $\mathcal{G} = (G = (Q, E), \lambda)$ such that $\mathcal{L}_{\mathcal{G}} = L(H_{\Gamma(k,d)})$. Let $N = |E|$.

Let $w = (R^{k-2}L^{k-2})^N R^k (L^2 R^2)^N$ be the ant-path described in Figure 5.4. It is a cycle of length $2Nk + k$. First, the ant follows semi-faces R^{k-2} and L^{k-2} . Then, it makes a complete tour of a face (R^k) . So, it is forced to return by the path $(L^2 R^2)^N$, i. e., any word coinciding with w in the first $2(k-2) + k$ letters and differing from w in some of the last letters does not belong to $L(H_{\Gamma(k,d)})$.

Let $u = (RL)^5$ be a zigzagging path of length 10. The word wLu is also an ant-path. There exists a sequence of edges of G : $(e_i)_{i=1}^{(2N+1)k+11}$ whose labels form the word wLu . After the cycle R^k in w , the ant is forced to follow the sequence $(L^2 R^2)^N$. The length of this sequence is four times the number of edges of G , N . So, there must be two repeated edges between $e_{2N(k-2)+k}$ and $e_{2N(k-1)+k}$, say $e_i = e_j$, with $2N(k-2) + k < i < j < 2N(k-1) + k$. Then, the sequence $\omega' = \lambda(e_1)\lambda(e_2)\dots\lambda(e_i)\lambda(e_{j+1})\dots\lambda(e_{(2N+1)k+11})$ must be a word of $H_{\Gamma(k,d)}$. It is a word of the form: $(R^{k-2}L^{k-2})^N R^k (L^2 R^2)^n vu$, where $n \leq N/2$ and v is a sub word of $L^2 R^2$.

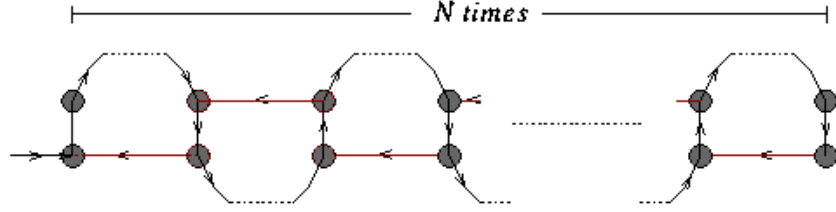


Figure 5.4: A cycle of length $2Nk + k$, corresponding to word w .

Then ω' does not contain the sequence $(L^2R^2)^N$ and so, it cannot belong to the language of $H_{\Gamma(k,d)}$, which is a contradiction. $\square$

Remark 3. *The former proof can not be applied when $k = 3$, because the word RL^2R is not an ant-path in $\Gamma(3,d)$. In fact, the former proposition requires the existence of cycles of arbitrary length, while in $\Gamma(3,d)$ their length is bounded by 6.*

Proposition 11. $(H_{\Gamma(k,d)}, \sigma)$, with $d \geq 5$ is a **coded system**.

Proof. Let us define the Fischer automaton $\mathcal{G} = (G = (Q, E), \lambda)$ as follows:

$$Q = \{w = w_1w_2\dots w_n \in L(H_{\Gamma(k,d)}) \mid \begin{array}{l} (n \geq 3) \text{ and} \\ (w_1w_2w_3 = RLR \text{ or } LRL) \text{ and} \\ (\forall i \leq n-3) w_iw_{i+1}..w_{i+3} \neq RLRL \text{ and} \\ (\forall i \leq n-3) w_iw_{i+1}..w_{i+3} \neq LRLR \end{array} \}$$

$$(u, v) \in E \iff \begin{array}{l} v = uL \text{ or} \\ v = uR \text{ or} \\ (v = LRL \text{ and } u = wRLR) \text{ or} \\ (v = RLR \text{ and } u = wLRL) \end{array}$$

$$\lambda((u, v)) = \begin{cases} L & \text{if } v = uL \text{ or } (v = LRL \text{ and } u = wRLR) \\ R & \text{in other cases} \end{cases}$$

It is direct that Q is countable.

(Q, E) is strongly connected. In fact, from transitivity, for any ant-path w , there exists an ant-path that begins with w and after a delay makes one of the sequences $RLRL$ or

LRLR. That is, any vertex w is connected to one of the vertices *LRL* or *RLLR*. *LRL* and *RLLR* are, obviously, connected to any vertex; any word in Q begins with *LRL* or *RLLR*. Which proves the connexity of the graph.

Finally, we shall prove that $L(H_{\Gamma(k,d)})$ is the language recognized by $\mathcal{G}$: To prove that $L(H_{\Gamma(k,d)}) \subset \mathcal{L}_{\mathcal{G}}$, for an ant-path $a = a_1..a_n$ it is enough to take the sequence of vertices defined by the sub-words $(a_{j_i-2}a_{j_i-1}a_{j_i}..a_i)_{i=1}^n$, where j_i is the maximal index less or equal than i such that $a_{j_i-3}..a_{j_i} = \textit{RLRL}$ or *LRLR*. If for some i there does not exist such an index, means that a does not begins with *RLLR* nor with *LRL*. In that case we take the smallest index $I \leq n - 1$ such that $a_{I-2}..a_{I+1} = \textit{RLRL}$ or *LRLR* and if it does not exists, we take $I = n$. By transitivity of $H_{\Gamma(k,d)}$ we know that given the ant-paths *LRL* and $a_1a_2..a_I$ there exists a word v such that the word $u = \textit{LRL}va_1..a_I$ is in $L(H_{\Gamma(k,d)})$. If u contains the word *LRLR* or the word *RLRL* (in an other place than the beginning), we consider a sub word of $u = u'a_l..a_I$ beginning with *LRL* or *RLLR* that does not contains *RLRL* nor *LRLR* in another place than the beginning and that finishes with $a_l..a_I$. In that way u belongs to V and if u' has a length larger than 2, the words $(u'a_1..a_i)_{i=1}^I$ are vertex in V and form a path whose labels form exactly the word $a_1..a_I$. If u' has length less than 2, let us suppose that $u' = L$ and consider the path in G : $(\textit{RLLR}, \textit{LRL}, u'a_l..a_{l+1}, \dots, u'a_l..a_I)$; it satisfies that its labels form the word $a_l..a_I$. The case $u' = R$ and u' an empty word, are analogous. Since $a_{I-2}..a_{I+1} = \textit{RLRL}$ or *LRLR* the path can be continued with the vertices $(a_{j_i-2}a_{j_i-1}a_{j_i}..a_i)_{i=I+1}^n$. So any $a \in L(H_{\Gamma(k,d)})$ is also in $\mathcal{L}_{\mathcal{G}}$.

To prove that $\mathcal{L}_{\mathcal{G}} \subset L(H_{\Gamma(k,d)})$ we use Theorem 5 of Chapter 4. Let $a \in \mathcal{L}_{\mathcal{G}}$, we must prove that there exists a configuration c such that $\psi(c) = x$ contains a . Since $a \in \mathcal{L}_{\mathcal{G}}$, there exists an infinite path $(v_i)_{i=1}^{n+1}$ of $G = (V, E)$ such that $(\forall i) a_i = \lambda((v_i, v_{i+1}))$. Let us consider the indexes i_j such that $v_{i_j} = \textit{LRL}$ or $v_{i_j} = \textit{RLLR}$. By definition of $\mathcal{G}$, the words $a_{i_j-3}..a_{i_{j+1}-1}$ are ant-paths. So we can define $s_0..s_{i_1-1}$ the set of cells that are visited by the ant in the path $a_0..a_{i_1-1}$. Figure 5.5 shows the cells $s_{i_1-3}..s_{i_1-1}$ when $v_{i_1} = \textit{RLLR}$ ($a_{i_1-1} = R$). By applying Theorem 5 to the ant in reverse time with respect to edge $\{s_{i_1-2}, s_{i_1-3}\}$ and the edge $\{s_{i_1-1}, s_{i_1-2}\}$ we know that cells $s_0..s_{i_1-2}$ are in the *gray* zone and that, consequently, the ant can turn to the left over s_{i_1} . So, the word $a_0..a_{i_1}$ is an ant-path. Applying the Theorem 5 with respect to edge $\{s_{i_1-1}, s_{i_1}\}$ and the edge $\{s_{i_1-2}, s_{i_1-1}\}$ we know that any ant's trajectory beginning with $s_{i_1-3}s_{i_1-2}s_{i_1-1}s_{i_1}$ will rest in the *dark gray* zone after iteration i_1 . Taking (s_{i_1-3}, s_{i_1-2}) as the initial position of the ant, we define $s_{i_1}..s_{i_2}$ the sequence of cells that are visited by the ant-path $a_{i_1-2}..a_{i_2}$ after iteration i_1 . So, we know that $\{s_0, \dots, s_{i_1-3}\}$ does not intersect $\{s_{i_1}, \dots, s_{i_2}\}$. Then $a_0..a_{i_2}$ is an ant-path. Recursively we prove that $a \in L(H_{\Gamma(k,d)})$.

□

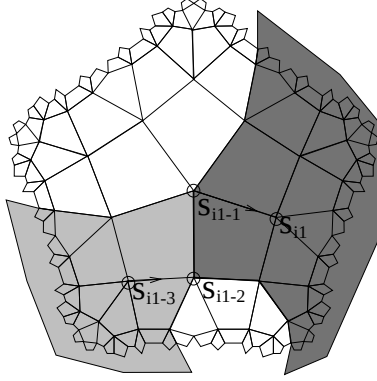


Figure 5.5: The ant's trajectory before i_{1-3} (light gray zone) does not intersect the ant's trajectory after i_1 (dark gray zone).

Proposition 12. $H_{\Gamma(3,d)}$ is a **Subshift of Finite Type** if $d \geq 6$.

Proof. The Fischer automaton $\mathcal{G} = (G = (Q, E), \lambda)$ that recognizes $H_{\Gamma(3,d)}$ is defined in Figure 5.6. We verify that the words of length 7 corresponds to a unique path in G .

To prove that it recognizes $H_{\Gamma(3,d)}$ we remark that the automaton establishes a rule to concatenate the four following words: $u_1 = R, u_2 = L, u_3 = RRLLLRL$, and $u_4 = LLRRRLR$. These words correspond to the paths showed in Figure 5.7. The words u_1 and u_2 are oscillations, the words u_3 and u_4 correspond to a displacement equivalent to those of words u_2 and u_1 respectively. We easily verify that we can concatenate these words as the graph of Figure 5.6(b) indicates, which implies that $\mathcal{L}_{\mathcal{G}} \subset L(H_{\Gamma(3,d)})$. For the other direction, the words not belonging to $\mathcal{L}_{\mathcal{G}}$ are the words that contain one of the following sequences: $LRRL, RLLR, RLRRRLR, LRLLLRL, LLLL$ or $RRRR$. All of them are not ant-paths in $\Gamma(3, d)$, they do not belong to $L(H_{\Gamma(3,d)})$. Which proves that $L(H_{\Gamma(3,d)}) \subset \mathcal{L}_{\mathcal{G}}$. $\square$

5.3 Towards universality

The modeling of the ant system proposed here can be applied also for Turing machines. In fact, the ant is nothing but a particular bi-dimensional Turing machine. The whole dynamics of a Turing machine can be decoded if we know the values of the tape symbols

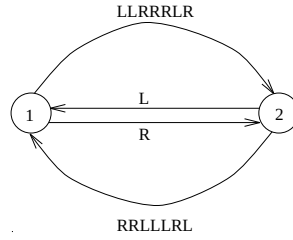


Figure 5.6: This automaton recognizes $H_{\Gamma(3,d)}$.

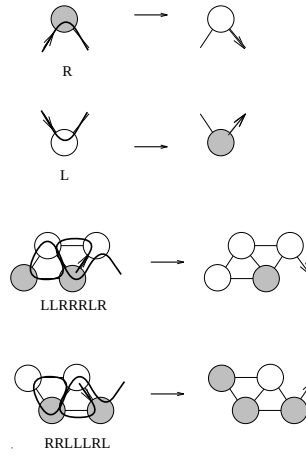


Figure 5.7: The paths corresponding to the automaton labels, the configurations that generate it and the configurations that they leave.

that the head reads and the internal states in the successive iterations. The following is one of the possible definitions of a Turing machine.

Definition 8. A **Turing Machine (TM)** is a 6-tuple,
 $M = (Q, S, \delta, q_0, \text{\textit{b}}, q_{stop})$, where Q and S are finite sets and

1. Q is the states set,
2. S is the input alphabet ($\text{\textit{b}} \notin S$),
3. $\delta : Q \times (S \cup \{\text{\textit{b}}\}) \rightarrow Q \times (S \cup \{\text{\textit{b}}\}) \times \{-1, 1\}$ is the transition function,
4. $q_0 \in Q$ is the starting state,
5. $\text{\textit{b}}$ is the blank symbol.
6. $q_{stop} \in Q$ is the halting state.

The machine works over a semi-infinite tape. Initially, an input word $w \in S^*$ is registered at the left of the tape, while the leaving cases in the tape are filled with the symbol $\text{\textit{b}}$. The machine can read one case at a time; and it can move its head just one case to left or to right. At the beginning the head is on the first case of the tape and its internal state is q_0 .

At each iteration, the machine reads a symbol and evaluates the δ as a function of its internal state and the read symbol. The result of this evaluation gives the next state, the new symbol of the current case, and a direction in which the heads moves on step.

When the machine achieves the state q_{stop} , it halts.

Now, we formulate a Turing machine as a dynamical system and we define the associated subshift.

Definition 9. Let $M = (Q, S, \delta, q_0, \text{\textit{b}}, q_{stop})$ be a Turing machine.

Let $B = (S \cup \{\text{\textit{b}}\})^* \times Q \times \mathbb{N}$ be the phase space of M .

The function $\Delta : B \rightarrow B$, is defined as follows:

Given a word $w = w_0 w_1 \dots w_m \in S^*$, a state $q \in Q$, and a head position $n \in \mathbb{N}$. The function δ defines the next state, the next move and the new symbol at position n :

$(a, q', d) = \delta(w_n, q)$. The next head position is $n + d$. If $n + d > m$ implies that $n = m$ and $d = 1$ then $w' = w_0 w_1 \dots a \text{ } \not b$. If $n + d \leq m$ then $(\forall i \neq n) w'_i = w_i$ and $w'_n = a$. The value of Δ on (w, q, n) is $(w', q', n + d)$.

The couple (B, Δ) is a formulation of the Turing machine as a dynamical system.

Let us define the projection function $\phi : B \rightarrow S \cup \{\not b\} \times Q$ by $\phi(w, q, n) = (w_n, q)$.

The subshift associated to (B, Δ) is the set

$$T = \{(\phi(\Delta^i(w, q, n)))_{i \in \mathbb{N}} \in ((S \cup \{\not b\}) \times Q)^{\mathbb{N}} \mid w \in S^*, q \in Q \text{ and } n \in \mathbb{N}\}$$

Proposition 13. *If the system T defined above, is sofic, then M cannot be universal.*

Proof. If T is sofic, then there is a Fischer automaton $\mathcal{G} = (G = (V, E), \lambda)$ such that $\mathcal{L}_{\mathcal{G}} = L(T)$.

We will prove that for any input word, M stops or it falls in a periodic dynamics. In other words, we will prove that for all $w \in S^*$, $(\phi(\delta^i(w, q_0, 0)))_{i \in \mathbb{N}_{set}}$ is eventually periodic.

Let w be an input word.

The following algorithm decides if M stops or not.

- Simulate M over w saving the current vertex over $\mathcal{G}$.
- When M has completely read w (we can suppose that this always happens). For each node in $\mathcal{G}$, with more than one outgoing edge: to erase all the edges with a label different from $\not b$.

Now, the new graph has degree 1. Then, its strongly connected components are cycles. To know whether M stops or not, it is enough to verify the presence of q_{stop} in the component of the remaining current nodes. $\square$

5.4 Discussion

As we said in Section 5.2, the soficity of (H_G, σ) (or (T, σ)), is associated to the length of the ant-cycles. More specifically, we can define the “ant’s memory” as the cells that the

ant has visited and that it can visit again. The size of this “memory” is directly related with the maximal length of ant-cycles. If the “ant’s memory” is finite, the (H_G, σ) is sofic. Proposition 11 says that a Turing Machine that has a “finite memory” cannot be universal, which is not surprising.

In the next chapter, we study the complexity of the ant, and we establish its universality in the case of the $\Gamma(k, 3)$ and $\Gamma(k, 4)$ graphs. Our interest in Proposition 11 is to relate the dynamical properties of H_G and its computing power. A trivial generalization of Proposition 11 to bi-dimensional Turing machines shows that *the ant over a $\Gamma(k, d)$ graph cannot be universal if $k = 3$* , considering the notion of simulation defined in the next chapter.

We think that soficity of (T, σ) implies that the length of the cycles that the head of the Turing machine can make is necessary bounded. Which implies that, T can be a subshift of finite type or a non sofic system but it cannot be sofic without being a subshift of finite type.

Chapter 6

Universality

In the former chapter we give a sufficient condition for a Turing Machine (TM) to be unable to be universal. More precisely, we prove that the halting problem is decidable over a certain class of TMs. This implies that these TMs cannot *simulate* every Turing machine. Otherwise, the halting problem becomes decidable on the whole set of TMs.

We say that a system is universal if it can simulate any TM, where simulate means to repeat, in a coded way, the behavior of a given TM on a given input. In that way, it reflects the regularities of the dynamics of the simulated TM. In particular we want that undecidability of the halting problem translates in the *existence of undecidable problems* on the simulating system.

We say “existence of undecidable problems” because the halting problem has no sense, in general, on a dynamical system. When a TM is formulated as a dynamical system, entering to the halting state implies attaining a fixed point in terms of dynamical systems.

In this Chapter¹ we face the problem of universality of the ant on the $\Gamma(k, d)$ graphs. First we show that the ant system is universal on the $\Gamma(k, d)$ graphs when $d = 3$ and $d = 4$. Our proof consist in showing how to transform the description of a Turing machine and an input word, into a configuration of the ant system. Starting from this configuration the ant simulates the Turing machine over the input.

In papers we find mainly two methods to show universality:

1 To directly simulate the tape and the head of a universal TM [43, 31].

¹The work presented in this chapter was developed in colaboration with Andrés Moreira

2 To simulate the gates and connections of a computer and through it, the behavior of a Turing machine [1].

Frequently we cannot simulate a computer with the needed flexibility to simulate with it a TM. We need to simulate other systems for which universality has already been proved. This is the case of two register machines [34, 17] and Cellular Automaton [16]. We use the second method. We simulate the behavior of a CA by computing logical circuits.

In [43] a CA is proved to be universal. The universal CA defined by Smith has a particular state called quiescent state. A quiescent state is defined as follows: if a cell and all its neighbors are in the quiescent state, the cell remains in it at the next iteration. Hence, all the dynamics of the system takes place at the cells in non-quiescent states and their neighbors. An finite initial configuration (i.e., with a finite number of non-quiescent states) will keep this property through the iterations of the CA. The simulation of Smith uses simulating configurations that are finite. This is important for us because in our construction we are not able to simulate a CA over an infinite configuration.

All the steps of our construction are developed in Sections 5.1.1, 5.1.2, 5.1.3 and 5.1.4. In Section 6.1.3 an additional result is given: When simulating logical circuits we reduce the problem of computing a logical circuit to a problem associated with the ants dynamics, establishing in that way, the existence of a P-hard problem on the ant system. Finally, in section 5.2 we discuss the existence of undecidable problems in the $\Gamma(k, d)$ graphs. In the case of $d = 3$ and $d = 4$ this is a consequence of universality. When $d \geq 5$ the decidability of a large gamma of problems is proved.

6.1 Universality in $\Gamma(k, d)$ graphs with $d = 3$ and $d = 4$

6.1.1 Construction of circuits

Choosing the initial configuration, we can impose a predetermined path to the ant (see Figure 6.1). In a regular graph of degree 3, we can force the ant to follow any simple path.

With this, we can make the ant to compute a logical gate. The idea is to represent the input and output variables by the states of certain cells. Figure 6.2 shows the idea of a logical gate. It is a sector of the graph, where we will define a configuration corresponding to a specific gate. At the top of the gate, we have some cells whose states

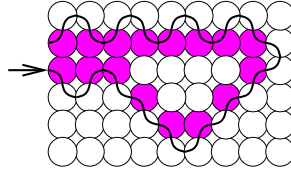


Figure 6.1: Forcing the ant to follow a path.

represent the input (the input cells). At the bottom, some cells represent the output (the output cells). Initially the output cells are in *to-left* state, which will represent the logical value false.

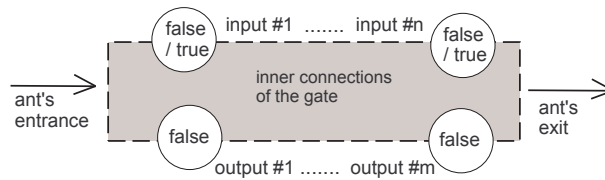


Figure 6.2: A sketch of a gate. The ant computes the logical gate by changing the states of the output cells. At the beginning the output cells are in *to-left* state, representing the logical value *false*.

The ant enters the gate from the left, follows some path inside the gate, and exits the gate by the right. While being in the gate, the ant visits the input cells and also some output cells. When it visits a cell it changes its state. The visited output cells will finish in *to-right* state. So, depending on the state of the input cells, the ant visits the correct output cells, according to the function which the gate represents.

The way to design the gates is represented in Figure 6.3. Let us describe how the NOT gate works. The ant visits the input cell, and depending on its state the ant will follow different paths. If the input cell is in *to-left* state (logical value false), the ant goes to the output cell and changes its state, and then exits. If the input cell is in *to-right* state (logical value true), the ant goes directly to the exit. This is the general scheme of the gates: the path of the ant bifurcates, depending on the input states. After (possibly) changing the output states, the paths are joined and the ant exits.

The changes are done *from inside*, thus allowing the output cell to be used as the input cell for another gate.

To compute a boolean circuit we just put the input variables in some cells at the top of

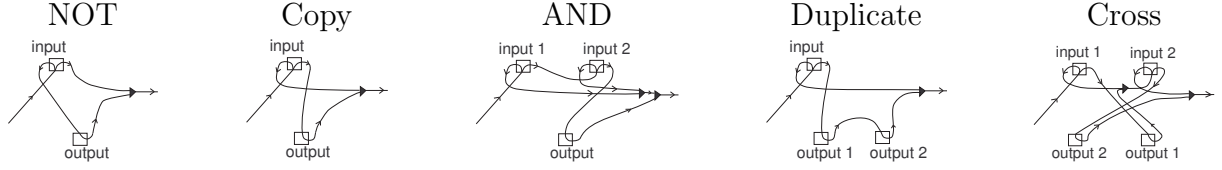


Figure 6.3: A simplified scheme of the gates.

the configuration (see Figure 6.4), and for the consecutive stages of evaluation we put consecutive rows of logical gates. The ant will go through every row, starting with the upper one. After going through the last row, the state of the last output cell will contain the evaluation of the circuit for the given input.

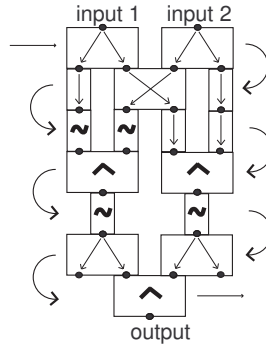


Figure 6.4: The XOR function, built upon AND, NOT, Crossing, Copy and Duplication gates. The ant computes row by row the logical circuit $(\sim (i_1 \wedge i_2) \wedge \sim (\sim i_1 \wedge \sim i_2))$. The circuit is satisfied if and only if the ant visits the output cell, for the given input.

To write a boolean circuit it is enough to have the NOT and the AND functions. To construct the circuit on the graph we also use gates that allow us to duplicate, cross and copy variables.

6.1.2 Embedding in infinite regular graphs

In the former section we show the method to simulate logical gates. In this section we show how to construct the configurations that performs this method.

First of all, we need to have paths for the ant to follow. As we said before, any path in a graph of degree 3 is an ant-path. The next lemma shows that in $\Gamma(k, 4)$, it is always possible to bring the ant from any location to any other location. We have seen in Chapter 4 that this simple fact is not true in the $\Gamma(k, d)$ graphs if $d \geq 5$.

Proposition 14. *Let $P = v_0, v_1, \dots, v_n$ be a simple path in $\Gamma(k, 4)$. Then there is a simple ant-path $A = a_0, \dots, a_m$ satisfying:*

- $(\forall 0 \leq i \leq m)(\exists 0 \leq j \leq n) a_i$ and v_j belongs to the same face,
- $a_0 = v_0, a_m = v_n$ and
- (a_{m-1}, a_m) and (v_{n-1}, v_n) belongs to the same face.

Proof. Let us describe the path by the relative position of consecutive edges: Left (L), Right (R), Opposite (O). So, we may write $P = (v_0, v_1; d_2 d_3 \dots d_n)$, where

$$d_i = \begin{cases} R & \text{if } (v_i, v_{i-1}) \text{ is to the right of } (v_{i-2}, v_{i-1}) \\ L & \text{if } (v_i, v_{i-1}) \text{ is to the left of } (v_{i-2}, v_{i-1}) \\ O & \text{if } (v_i, v_{i-1}) \text{ and } (v_{i-2}, v_{i-1}) \text{ don't belong to the same face} \end{cases}$$

This is well defined, since each node has degree 4, and the path is simple. In this context, an ant-path is described by a sequence that does not contain the symbol O.

We will prove the Proposition by induction over the length of P . If $n = 2$, P consists of 2 edges. If they belong to the same face, then P is an ant-path. If not, then let us define $A = (v_0, v_1; LR^{k-2})$. In Figure 6.5 we can see that A satisfies the Proposition.

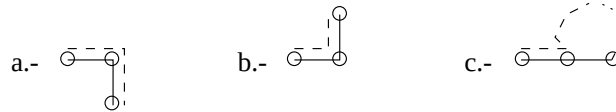


Figure 6.5: Case $n = 2$. The solid line stands for P , the dashed one for the ant-path A .

Now let's suppose that the Proposition is true for any path of length n . Let P be a path of length $n + 1$: $P = (v_0, v_1, \dots, v_{n+1})$, and let $A' = (a_0, a_1, \dots, a_m)$ be the ant-path that fits $(v_0, v_1, \dots, v_n)$. There are several cases:

- $a_{m-1} = v_{n-1}$. Then we may define $(a_{m-1}, a_m, \dots, a_{m+r})$ as the ant-path that fits (v_{n-1}, v_n, v_{n+1}) , and we take $A = (a_0, a_1, \dots, a_{m+r})$.
- $a_{m-1} \neq v_{n-1}$. There are three cases:
 1. $v_{n+1} = a_{m-1}$. Then we take $A = (a_0, a_1, \dots, a_{m-1})$.
 2. (v_n, v_{n+1}) and (a_{m-1}, a_m) belong to the same face. Then we take $A = (a_0, a_1, \dots, a_m, v_{n+1})$.
 3. (v_n, v_{n+1}) and (a_{m-1}, a_m) don't belong to the same face. Let us define $(a_{m-1}, a_m, \dots, a_{m+r}) = (a_{m-1}, a_m; LR^{k-2})$ if (v_{n-2}, v_{n-1}) is to the right of (a_{m-1}, a_m) , and $(a_{m-1}, a_m, \dots, a_{m+r}) = (a_{m-1}, a_m; RL^{k-2})$ otherwise. Now we make take $A = (a_0, a_1, \dots, a_{m+r})$.

Since the new path A is described only by using the symbols R and L, it is a good candidate to be the needed ant-path; we just have to verify that it is simple. By induction hypothesis, $(a_0, \dots, a_m)$ is simple. If $(a_0, \dots, a_{m+r})$ is not simple, then there exist $i \in \{0, \dots, m\}$ and $j \in \{m+1, \dots, m+r\}$ such that $a_i = a_j$. Let us choose i and j so that $j - i$ is as large as possible. If $j = m + r$, we re-define $A = (a_0, \dots, a_i)$, and A satisfies the Proposition. In the other case, let us define $A = (a_0, \dots, a_i, a_{j+1}, \dots, a_{m+r})$. A is an ant-path, as Figure 6.6 shows, and it satisfies the Proposition.

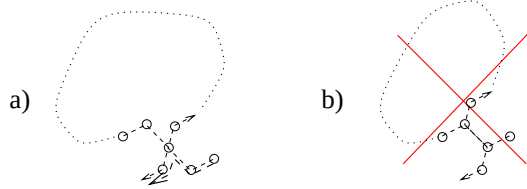


Figure 6.6: a) The short-cut path turns to the left; it is an ant-path. b) If this happens, it implies that the ant-path passes over an edge in both directions, but this is impossible in a graph of even degree (Section 3.1).

□

The sketches of Figure 6.3 suppose that we can cross and join two paths. To do that, we define configurations for a finite sector of the graph that can be inserted at the places where we need crossings or junctions. With reference to Figure 6.7, they satisfy:

- **Crossing:** If the ant first enters at 1, it exits at 2. If afterwards it enters at 3, it exits at 4. But if it enters first at 3, it exits at 5.

- **Junction:** If the ant enters by 1 or 2, it exits by 3.

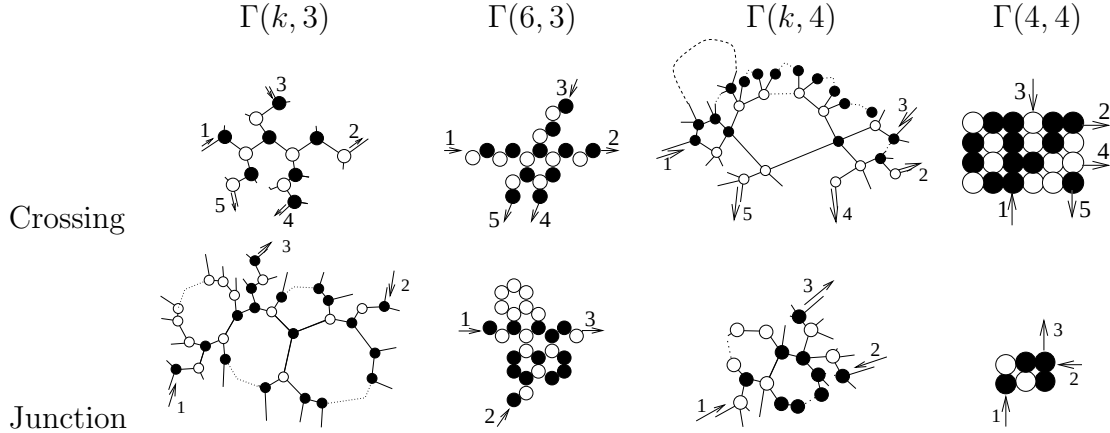


Figure 6.7: White stands for *to-left*, black for *to-right*. Dashed lines stand for simple paths that do not intersect the rest of the configuration.

Following Figure 6.3 and using the configurations of Figure 6.7, and simple paths, we define configurations that simulate the AND, NOT, Cross, Copy and Duplicate gates. We can choose the dimensions of these gates and the positions of their inputs and outputs arbitrarily, and this can be done in an automatic way. A procedure that takes a boolean circuit and writes the corresponding configuration in a $\Gamma(k, d)$ graph can thus be defined.

Figure 6.8 shows a Duplicate gate for $\Gamma(4, 4)$ and $\Gamma(6, 3)$ (the square and the hexagonal lattices, respectively). It is not possible to show a gate in a hyperbolic graph; the fast grow of the graph allows to show only a small part of the graph. A configuration used to calculate a logical gate for a given input cannot be used any more, the path are changed they do not satisfies the same condition. This justify the staking of gate to compute a bigger circuit

6.1.3 Computational Complexity

The problem (CIRCUIT-VALUE) of determining, given a boolean circuit C and a truth assignment t , whether C outputs *true* on input t , is known to be P -complete ([36]). Now, let us fix (k, d) , with $d = 3$ or 4 . From 6.1.2, for any pair (C, t) we can build a configuration in $\Gamma(k, d)$ representing them, so that the ant will end the last row having

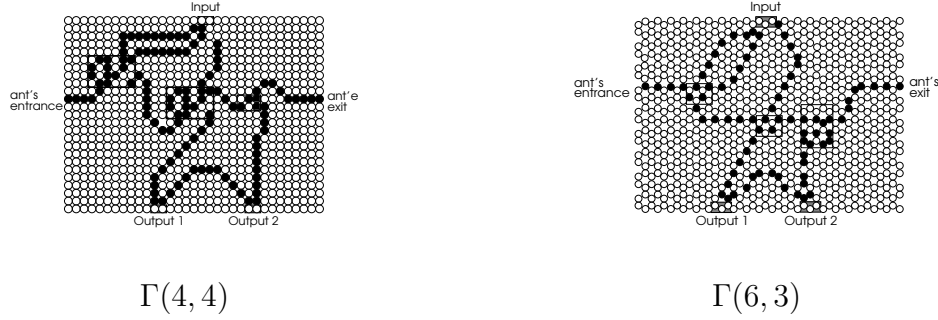


Figure 6.8: *Duplicate* gate in the square and hexagonal lattices.

visited or not the output node of that row, depending on the outcome of C with input t . Thus the problem (CIRCUIT-VALUE) is being reduced to the problem (P) of knowing, for a finite initial configuration of $\Gamma(k, d)$, whether the ant visits a given node α before another given node β , or not.

This reduction is logarithmic: the number of gate's rows is bounded by two times the height (H) of C plus the number of crossings, i.e., $2(H + W^2H)$, where W is the width of C . The number of gates in each row is bounded by W . That implies that the number of *to-right* cells necessary to simulate a size $H \times W$ circuit is bounded by $2SWH(W^2 + 1) = o(W^3H)$, where S is the number of *to-right* cells necessary to simulate a single gate.

The algorithm that defines the simulating configuration needs only logarithmic space; all it has to do is to read and translate the boolean circuit. For this purpose, it has to memorize numbers such as the position of the symbol that is being translated and the current height of the circuit; these numbers are bounded by the size of the input, and can be recorded in logarithmic space. The output of this “drawing” algorithm may be the list of coordinates of cells in the *to-right* state, which is polynomial in the length of the input. This is all we need to legitimate the reduction, and we conclude that (P) is P -hard.

6.1.4 Conclusion

The transition rule of a CA can be calculated with a multi-output finite boolean circuit. So, for a given one-dimensional CA with quiescent state and a number L , we can define an initial configuration on the $\Gamma(k, d)$ graph consisting of L copies of this circuit, arranged in a row. Any initial configuration of the CA whose support has width less than L

can be written as the input of this row of circuits, and the ant will calculate the next configuration of the CA. A next row of copy and cross gates makes the ant to reorganize this configuration as the input of a next row of circuits of length $L + 2$. So, stacking infinitely many rows of growing length as shown in Figure 6.9 the ant simulate the whole evolution of the CA. For widths bigger than L , just put the initial configuration in a lower row, and let the ant start running from the appropriate node.

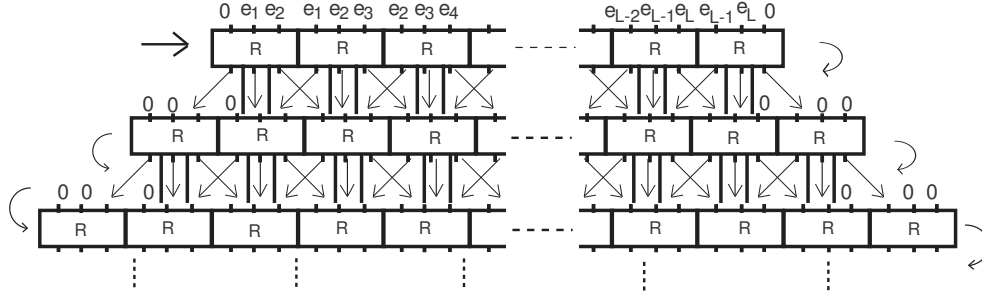


Figure 6.9: The ant simulates each iteration of the CA in a row of gates, it crosses the repetitions of the outputs (preparing the next input) and goes to the next row. R stands for the circuit that computes the rule.

Since any Turing Machine can be simulated by a 1D CA with a finite initial configuration, the ant is able to simulate any TM. The codification is the following: Let us define R as the logical circuit that computes the transition function of a CA that simulates a Universal TM: UT . Let us consider a fixed integer L and define the configuration described in Figure 5.9. Then, given an arbitrary TM and an input word, let us consider its respective codifications which are the inputs of UT . Afterwards consider the codification of these (that is the initial configuration of the universal CA defined by Smith). Consider the binary coding of the states of the CA and code its initial configuration accordingly. We obtain a binary word say b , let K be its length. Define the states of the input cells of a row in the simulating logical circuit having a width larger than K , as b . Now, if we leave the ant to evolve, we will see how the successive iterations of T over w are emulated in a coded way.

The configuration that simulates a CA is finitely described, because it is a regular repetition of a finite pattern. We can change a little bit this configuration to makes it periodic and in that way to prove that *there exist undecidable problems concerning the ant's evolution on periodic initial configurations*. We modify the circuit R in the previous construction in the way shown in Figure 6.10: When the ant exits from R , it visits a cell that we call "stop mark"; if it is in *to-left* state the ant continue into the

next circuit R , otherwise it follows a path that goes to put the "stop mark" in a farther and lower block R . After doing this the path joins the entry of the cross and copy row.

The cross and copy row is also provided with "stops marks". When the ant exits a cross, it visit the "stop mark". If it is in *to-left* state it continues. If it is in *to-right* state the ant follows a path that carries it to put a "stop mark" in the next row of cross one block farther to the left. After this it comes back to compute the next row of logical circuits and so on. At the beginning we put "stops marks" at both sides of the input, and these are copied by the ant in the following rows. This modified configuration is periodic except for a finite number of cells: the input word and the two stops marks (see Figure 6.11).

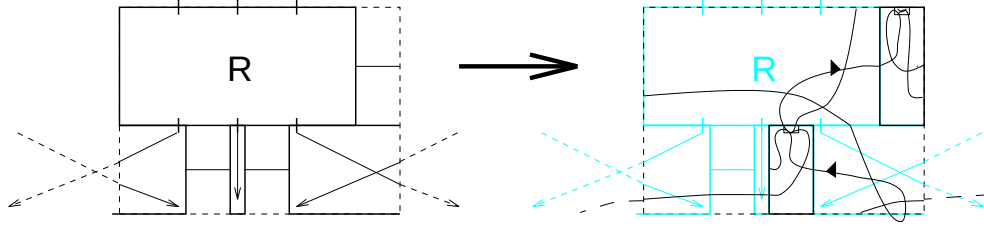


Figure 6.10: The block given here defines an infinite, periodic configuration.

6.2 Decidability

The halting problem translates as the apparition of a certain pattern (the pattern that codes the halting state in the output of a circuit $\mathbf{R}$) on a configurations c_t at some iteration of the ant. This shows the existence of undecidable problems associated to infinite configurations on the $\Gamma(k, d)$ graphs with $d = 3$ or 4 . This problem is associated to an infinite but periodic initial configuration. We do not know the existence of undecidable problems associated to finite initial configurations. Finding such a problem shall answer negatively the conjecture that says that the highway always appears when starting from a finite configuration. In fact, if the conjecture is true, any problem associated to a finite initial configuration becomes decidable, for in that case, it is enough to iterate on the configuration until the regular trajectory appears. The question may be answered at that point, since the future dynamics is easily predicted.

In Chapter 4 we have proved that the ant's trajectory is eventually regular for any finite initial configuration if the underlying graph is an infinite $\Gamma(k, d)$ and $d \geq 5$. That is,

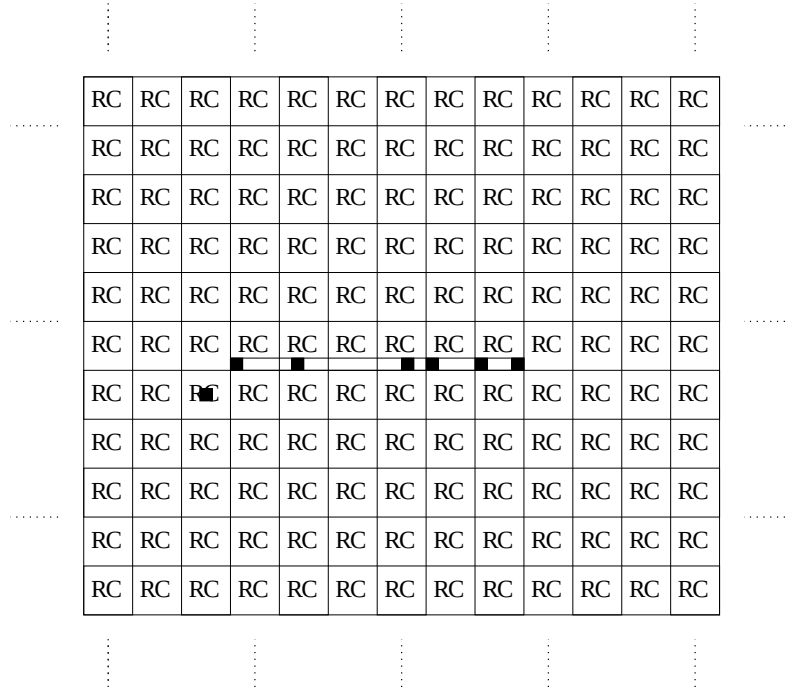


Figure 6.11: Each block RC contains the logical circuit R and the cross. The ant travel through each row in both directions. From left to right it computes the circuits R and from right to left it makes the cross. We add “stop marks”, at right and left of the row that contains the ant and the initial configuration of the CA. When the ant finds a mark, it copies it to the next row, and comes back until it finds another mark and repeats the operation. The parametric input consists in the first states and the two ”stop marks”.

the conjecture is true on the graphs of degree ≥ 5 . So, any problem associated to a finite initial configuration is decidable. We think that problems associated to periodic configurations are also decidable. To prove such a thing, it is necessary to define a notion of periodicity in the tree of faces to which the ant is restricted.

From the analysis made at the beginning of this chapter, the result of Chapter 4 allows us to say that the ant cannot be universal if we require the simulating configuration to be finite. If we want to simulate an arbitrary Turing machine in the ant system when the underlying graph is a $\Gamma(k, d)$ and $d \geq 5$, then we must consider only infinite configurations. Nevertheless, the configuration defined to prove universality in Section 6.1 cannot be embedded on the $\Gamma(k, d)$ with $d \geq 5$. To prove universality on these graphs, a completely different method should be necessary.

pointing in some direction. The cell state is not associated now with a turn direction. It is only a color that will affect the ant position and the pointing direction: it can change or not its pointing direction, and it can makes the ant to go backward or forward.

To obtain a major diversity of behaviors, we allow the ant also to not change the cell state in some cases. We define in that way several different rules for the ant. Each rule is described by a 2×2 matrix as follows:

	color change	ant's movement
black cell	a	c
white cell	b	d

Where $a, b \in \{0, 1\}$, 1 meaning that the ant changes the color of the cell; $c, d \in \{0, 1, 2\}$ codify the ant's movement. The movements are described in the following table. The first row indicates the initial ant position, the next rows shows the next ant positions for the different kinds of movements.

initial position		$\leftarrow$	
0	$\leftarrow$		
1			$\rightarrow$
2	$\rightarrow$		

0 means that the ant moves forward and maintains its heading direction.
1 means that the ant moves backward, and changes its heading direction.
2 means that the ant moves forward and changes its heading direction.

This definition is invariant under reflection. If the ant heads initially to the right, the movements are defined analogously.

For example, the rule $\begin{pmatrix} 10 \\ 11 \end{pmatrix}$ produces the dynamics showed in Figure 7.2.

In this context there are 36 possible rules, however 12 rules are independent of the cell color and its behavior is simple (the ant propagates indefinitely or it cycles with period two between two contiguous cells). Then, we only need to study 24 rules. It is direct that when we interchange the rows of the matrix of some rule, the new rule have the same behavior by interchanging the colors. We call them *dual rules* and they have analogous properties. We will make the proof of the different properties only for one rule of the dual couple.

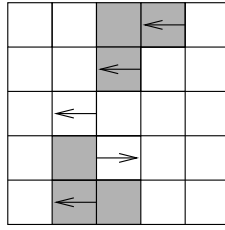


Figure 7.2: A simulation of rule $\begin{pmatrix} 10 \\ 11 \end{pmatrix}$. The initial configuration is at the top. The configurations for iterations 1 to 4 are ranged below.

Let us notice that the rule $\begin{pmatrix} 10 \\ 11 \end{pmatrix}$ was already studied in [25], where statistical properties are studied analytically.

We classify the rules first in two classes: always unbounded and bounded for some initial configurations. Each class is subdivided in sub classes depending on the allowed periods (in the case of the bounded ones), and its steady state when starting over a finite configuration. We found only three kinds of steady state trajectories: periodic, constant velocity movement and oscillations with always growing amplitude (see Figure 7.3).

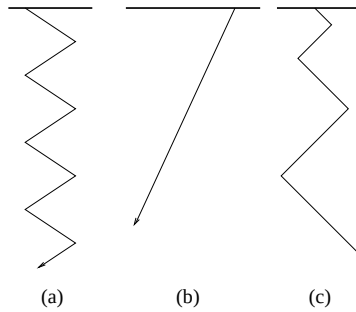


Figure 7.3: Steady state trajectories: (a) Bounded or periodic (b) Unbounded to the left (c) Unbounded covering every cell in $\mathbb{Z}$

We define the following classes of rules:

B : The set of rules that admit bounded trajectories.

$B1$: the rules in B that admit only periods of length 2 or 4.

$B2$: the rules in B that admit periods of arbitrary length.

U : The set of unbounded rules.

$U1$: The rules in U such that for some finite initial configuration the ant's trajectory is eventually regular.

$U2$: The rules in U such that for no finite initial configuration the ant's trajectory is eventually regular, and the ant visits all the cells in $\mathbb{Z}$.

These 4 classes cover all the considered rules. No other behaviors are possible with the rules that we have chosen.

7.1 Class B

This class is characterized because for each rule there exist a pattern that makes the ant to propagate constantly in a fixed direction. When propagating, the ant modify this pattern and when it exists this pattern, it falls in a bounded behavior (the exception is for rule $\begin{pmatrix} 00 \\ 01 \end{pmatrix}$ that does not modify the configuration at all). The ant will have an unbounded trajectory if the pattern that makes it to propagate is infinite in the direction that the ant points. The rules belonging to B are given in Table 7.1.

Bounded rules					
B1				B2	
$\begin{pmatrix} 01 \\ 02 \end{pmatrix}$	$\begin{pmatrix} 02 \\ 01 \end{pmatrix}$	$\begin{pmatrix} 10 \\ 02 \end{pmatrix}$	$\begin{pmatrix} 02 \\ 10 \end{pmatrix}$	$\begin{pmatrix} 00 \\ 01 \end{pmatrix}$ $\begin{pmatrix} 01 \\ 00 \end{pmatrix}$	
$\begin{pmatrix} 11 \\ 02 \end{pmatrix}$	$\begin{pmatrix} 02 \\ 11 \end{pmatrix}$	$\begin{pmatrix} 10 \\ 01 \end{pmatrix}$	$\begin{pmatrix} 01 \\ 10 \end{pmatrix}$		
$\begin{pmatrix} 01 \\ 12 \end{pmatrix}$	$\begin{pmatrix} 12 \\ 01 \end{pmatrix}$	$\begin{pmatrix} 00 \\ 02 \end{pmatrix}$	$\begin{pmatrix} 02 \\ 00 \end{pmatrix}$		

Table 7.1: Rules belonging to class B .

We sub-classify this class in subclasses B1, B2, in function of its specific behavior. We distinguish class B2 from class B1 because the ant with a rule in B2 may reach any even-cycle while with the rules in class B1 it may reach only two and four-cycles.

7.1.1 Class B1

In the next lemma we analyze the rules that are in the first two columns of Table 7.1. They have a very similar behavior so one sole proof will be done for all of them. They propagate over the same pattern which is a sequence of cells whose colors alternates.

Lemma 3. *Given a finite sequence of cells with alternating colors and an ant on the right of the sequence as follows: $\square$ or $\blacksquare$. Then, the ant evolving under one of the rules $\begin{pmatrix} 01 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 11 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 01 \\ 12 \end{pmatrix}$, $\begin{pmatrix} 11 \\ 12 \end{pmatrix}$, will advance at speed 1, until it leaves the sequences. If the ant begins on the left of the sequence as follows: $\square$ or $\blacksquare$, then, it will advance to the right.*

Proof. To prove this Lemma it is enough to simulate the ant under the conditions of the hypothesis. Rules $\begin{pmatrix} 01 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 11 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 01 \\ 12 \end{pmatrix}$ and $\begin{pmatrix} 11 \\ 12 \end{pmatrix}$ have the same movement just differ on the color that they yields. Its dynamics can be seen in Figure 7.4.

□

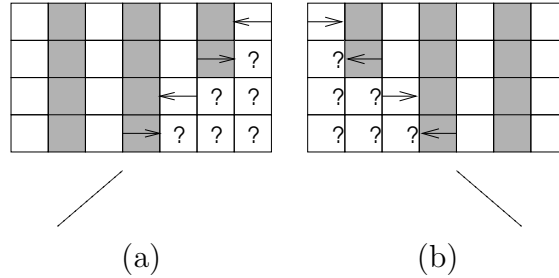


Figure 7.4: Evolution over a sequence of alternating colors. (a) To the left. (b) To the right. Symbol ? codes black or white color, depending of the specific rule.

Lemma 4. *Consider a rule in anyone of the first two columns of Table 7.1 and an arbitrary initial configuration. Then, when an ant finds for the first time two consecutive cells in the same color, it keeps into two cells, oscillating with period 2.*

Proof. We will analyze the rules $\begin{pmatrix} 01 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 11 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 01 \\ 12 \end{pmatrix}$. It is direct that when an ant enters by the first time into a block of cells in the same color it comes from a cell of the opposite color. Suppose first that it enters a block of black cells. Without loss of generality suppose the ant enters from the right, so the only possible configuration is as follows



Analyzing this situation for every rule we conclude that in at most two steps the ant reaches a two-cycle. For a block of white cells, we have a similar behavior (see Figure 7.5). $\square$

The ant entering in a black-block			The ant entering in a white-block		
Rule $\begin{pmatrix} 01 \\ 02 \end{pmatrix}$	Rule $\begin{pmatrix} 11 \\ 02 \end{pmatrix}$	Rule $\begin{pmatrix} 01 \\ 12 \end{pmatrix}$	Rule $\begin{pmatrix} 01 \\ 02 \end{pmatrix}$	Rule $\begin{pmatrix} 11 \\ 02 \end{pmatrix}$	Rule $\begin{pmatrix} 01 \\ 12 \end{pmatrix}$
...	...	...	...	...	...

Figure 7.5: Two-cycle evolution of rules $\begin{pmatrix} 01 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 11 \\ 02 \end{pmatrix}$, $\begin{pmatrix} 01 \\ 12 \end{pmatrix}$ when an ant enters a block of cells in the same color.

We conclude that the ant with these rules advances over an alternating pattern but it “stops” when it finds two cells with the same color.

Now let us study the other rules in B1.

Lemma 5. *The ant’s trajectory under rules $\begin{pmatrix} 10 \\ 02 \end{pmatrix}$ and $\begin{pmatrix} 10 \\ 01 \end{pmatrix}$ admits only cycles of length 2 and under rule $\begin{pmatrix} 00 \\ 02 \end{pmatrix}$ it admits only cycles of length two or four.*

Proof. Let us analyze first the rule $\begin{pmatrix} 10 \\ 01 \end{pmatrix}$. Suppose that the ant is over a white cell. If the cell behind it is also white, the ant oscillate between this two cells. If the cell behind it is black, the ant advances over the black cells until reaching another white cell, and it yields all the visited cells in white state. Thus finding the first case.

If there is no white cells in the direction that the ant points, the ant will propagate indefinitely in this direction. In other case, the ant will fall in a periodic trajectory with period 2.

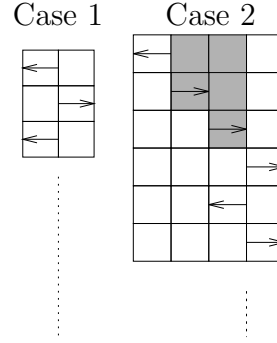


Figure 7.6: Evolution of rule $\begin{pmatrix} 10 \\ 01 \end{pmatrix}$ with departure in a white cell.

The analysis of rule $\begin{pmatrix} 10 \\ 02 \end{pmatrix}$ is similar to the previous one and the ant always reaches a cycle of length 2.

Let us study now the rule $\begin{pmatrix} 00 \\ 02 \end{pmatrix}$. Suppose that the departure cell is white. As Figure 7.7 shows, the ant always enters in a cycle. The period depends on the color of the neighboring cells.

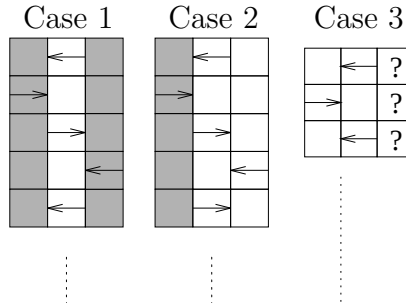


Figure 7.7: Four and two cycles behavior of rule $\begin{pmatrix} 00 \\ 02 \end{pmatrix}$ for a white departure cell.

We conclude that if the ant finds a white cell, it falls in a two or four-cycle. If not, it advances constantly in a fixed direction and its trajectory is unbounded. $\square$

7.1.2 Class B2

Lemma 6. *For rule $\begin{pmatrix} 00 \\ 01 \end{pmatrix}$, we have three situations.*

1. If the ant's cell departure is in a sector of black cells, then it oscillates with a period of two times the sector diameter + 2.
2. If the ant's cell departure is in a white cell with a sector of black cells behind, the ant has the behavior described in 1.
3. If the ant's cell departure is in a white cell with a white cell behind, then the ant oscillates between the two-white cells with period two.

Proof. It is enough to see the simulation of the ant on the three previous situations in Figure 7.8. One observes that situation 2 is a part of the behavior of situation 1.

We can observe that the rule $\begin{pmatrix} 01 \\ 00 \end{pmatrix}$ is unbounded over all the initial configurations where the ant is not enclosed by black cells.

□

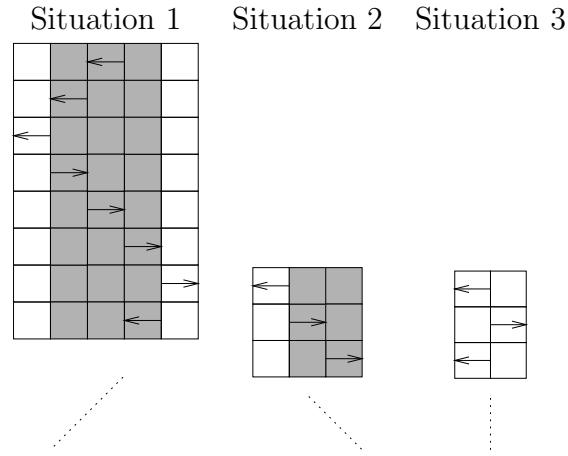


Figure 7.8: Ant's evolution under rule $\begin{pmatrix} 00 \\ 01 \end{pmatrix}$. (Situation 1) Departure in a black cell. (Situation 2) Departure in a white cell and a black cell behind. (Situation 3) Departure in a white cell and a white cell behind.

We conclude that for each rule there exist some initial configuration that produces a bounded trajectory on the ant.

Figure 7.9 shows examples of the ant's trajectory in this class.

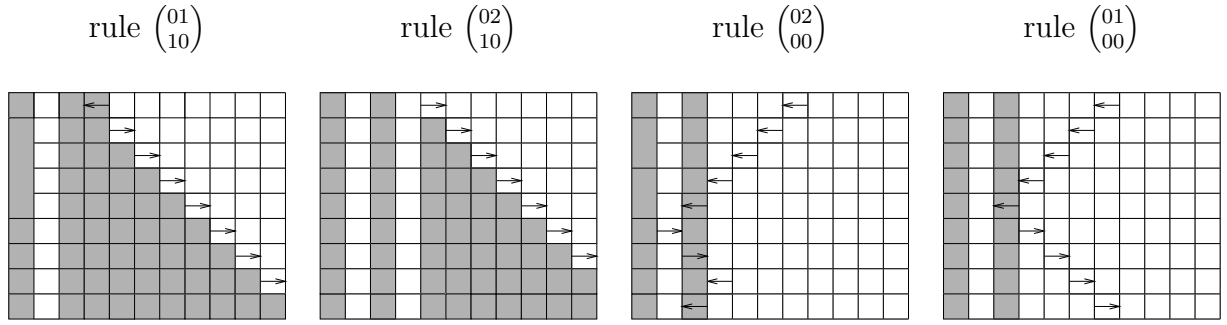


Figure 7.9: Examples.

7.2 Class U

The rules that are not in Class B are in Class U. The ant's trajectory is always unbounded with this rules. This is, in general, because the ant propagates in a regular way over some pattern, but when it finds a perturbation, it does not “stops,” but it rebounds or it over comes the perturbation. If this pattern is homogeneous, the ant is in class $U1$, because when the ant attains the homogeneous part of a finite configuration, its trajectory becomes regular. If the pattern is not homogeneous, the ant oscillates indefinitely and visits all the cells of the graph if it starts over a finite configuration.

Unbounded rules				
U1				U2
$\begin{pmatrix} 11 \\ 10 \end{pmatrix}$	$\begin{pmatrix} 12 \\ 00 \end{pmatrix}$	$\begin{pmatrix} 12 \\ 10 \end{pmatrix}$	$\begin{pmatrix} 11 \\ 00 \end{pmatrix}$	$\begin{pmatrix} 11 \\ 12 \end{pmatrix}$
$\begin{pmatrix} 10 \\ 11 \end{pmatrix}$	$\begin{pmatrix} 00 \\ 12 \end{pmatrix}$	$\begin{pmatrix} 10 \\ 12 \end{pmatrix}$	$\begin{pmatrix} 10 \\ 12 \end{pmatrix}$	$\begin{pmatrix} 12 \\ 11 \end{pmatrix}$

Table 7.2: Rules belonging to class U.

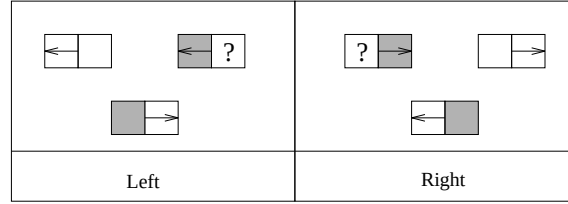
7.2.1 The class U1

This class is composed by eight rules that are showed in Table 7.2. We shall prove that each one of them is unbounded for any initial configuration. A proof will be made for

each rule (and its dual one), since in each case the arguments are different. The fact that the ant may have a regular trajectory starting with a finite configuration will be seen easily from the analysis of each rule.

Lemma 7. *The ant with rule $\begin{pmatrix} 11 \\ 10 \end{pmatrix}$ or $\begin{pmatrix} 10 \\ 11 \end{pmatrix}$ propagates in the same direction all the time. The direction depends exclusively on the color of the cell where the ant starts and the color of the cell behind it.*

Proof. For the rule $\begin{pmatrix} 10 \\ 11 \end{pmatrix}$, the following figure shows the situations in which the ant goes to the left and the ones in which the ant goes to the right. They cover all the possibilities, and divide them into two groups.



Indeed, if the ant begins in a situation of the Left group, then, at the next step it will continue in the Left group. Furthermore, after one or three steps it will move at least one cell to the left. Then it propagates to the left with velocity at least $1/3$. Figure 7.10 shows simulations of the ant rule on the situations of the Left group.

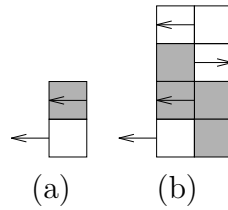


Figure 7.10: (a) If the ant is over a black cell, in the next step it will be one cell to the left, with a white cell behind. (b) If the ant is over a white cell with a white cell behind, after three steps it will be one cell to the left, with a white cell behind.

The analogous happens if the ant begins in a situation of the Right group.

□

Figure 7.11 shows an example of the behavior of rule $\begin{pmatrix} 10 \\ 11 \end{pmatrix}$. Let us notice that the movement of the ant can be separated in boxes where the cell at bottom-left has always the ant on a black cell, pointing to the left.

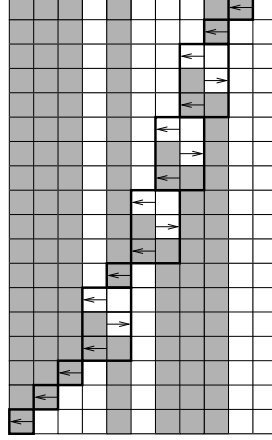


Figure 7.11: A simulation of the rule $\begin{pmatrix} 10 \\ 11 \end{pmatrix}$.

Lemma 8. *The rule $\begin{pmatrix} 00 \\ 12 \end{pmatrix}$ or $\begin{pmatrix} 12 \\ 00 \end{pmatrix}$, has an unbounded behavior over any initial configuration.*

Proof. We will prove the Lemma only for the rule $\begin{pmatrix} 12 \\ 00 \end{pmatrix}$.

Let us suppose that it is not true. Let c be an initial configuration such that the rule is bounded over it. Therefore sooner or later it will fall in a periodic behavior.

Let us consider a sector of the space, say $P \subset \mathbb{Z}$, that the ant visits periodically and let α be the cell that is leftmost in P .

From the first time the ant visits α , it becomes white. Figure 7.12 shows what happens when the ant visit α in white state: (a) it visits α pointing toward left, then it goes out P . (b) it visits α pointing toward right, then it must come from a black cell. But that can happen just one time because after that the cell to the right of α will be white forever. Then the rule can not be bounded on c . $\square$

The behavior of rule $\begin{pmatrix} 12 \\ 00 \end{pmatrix}$ is summarized as follows (see Figure 7.13):

- If the ant is in a white cell, it advances until falling in a black cell.

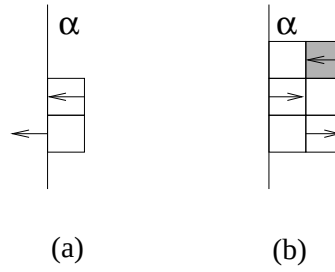


Figure 7.12: Just (b) is possible, but for the next time $\alpha + 1$ will not be black.

- If it is in a black block of cells, it advances until finding a white cell, leaving the visited cells in white color. If it advances an odd number of cells, it returns. If it advances an even number of cells, it continues in the same direction.

The ant changes direction only when it passes from a black block to a white block. To pass by all the cells of the space it is necessary to change infinite times of direction, then, it is necessary an infinite number of white and black cells. We conclude that for rules $\begin{pmatrix} 00 \\ 12 \end{pmatrix}$ and $\begin{pmatrix} 12 \\ 00 \end{pmatrix}$ the ant visits only a half-space when it begins in a finite support initial configuration.

Lemma 9. *The ant's trajectory is always unbounded with the rule $\begin{pmatrix} 10 \\ 12 \end{pmatrix}$ and $\begin{pmatrix} 12 \\ 10 \end{pmatrix}$.*

Proof. We will prove the Lemma only for the rule $\begin{pmatrix} 10 \\ 12 \end{pmatrix}$.

Let us suppose that it is not true. Let c be an initial configuration such that the ant's trajectory is bounded over it. Therefore sooner or later it falls in a periodic behavior.

Let us consider a sector of the space, say $P \subset \mathbb{Z}$ that the ant visits periodically and let α be the cell that is leftmost in P .

Each time the ant visits α it changes its color. Figure 7.14 shows what happens when the ant visits α and it is white.

In both cases the ant falls in a cell to the left of α , that contradicts the fact that α is the last cell to the left for where the ant passes periodically. $\square$

As before, the ant changes direction only when it passes from a white cell to a black cell. If the number of black cells is finite, the ant oscillates a finite number of times,

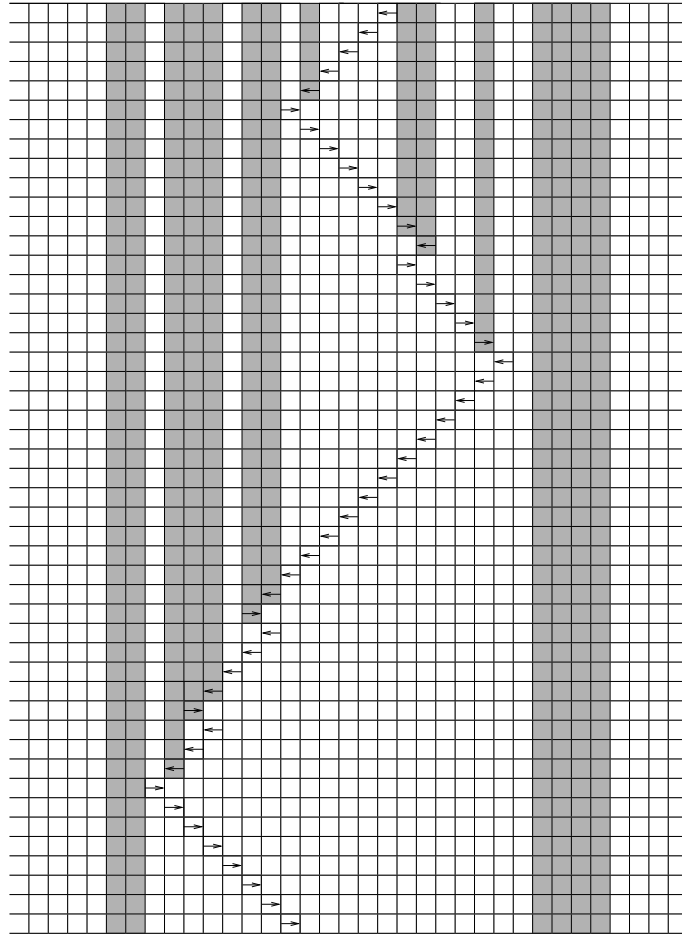


Figure 7.13: A simulation of the rule $\begin{pmatrix} 12 \\ 00 \end{pmatrix}$.

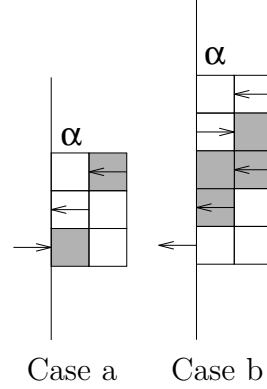


Figure 7.14: In any case the ant exits from P .

afterwards it advances constantly to infinite. See Figure 7.15 as an example.

Lemma 10. *The ant's trajectory is always unbounded with rules $\begin{pmatrix} 11 \\ 00 \end{pmatrix}$ and $\begin{pmatrix} 00 \\ 11 \end{pmatrix}$.*

Proof. Let us suppose that they have a bounded behavior. Then, there is a sector of the space, say $P \subset \mathbb{Z}$, where the ant passes periodically. Now consider a cell α in the left border of P . Since the ant visits α infinite times, and from the first time it changes it to white color (black in the case of the rule $\begin{pmatrix} 00 \\ 11 \end{pmatrix}$), let us study the case where α is in white color. If it passes pointing toward out, the transition rule indicates that it will go to the left, contradicting the assumption. And it cannot pass pointing toward inside, because in that case it must come from the left. Then the rule must be unbounded. $\square$

With the rule $\begin{pmatrix} 11 \\ 00 \end{pmatrix}$ the ant changes of direction only when it passes over a black cell. Then if there are a finite number of black cells, the ant oscillates a finite number of times and it visits just a half-space. When starting over a configuration with a finite number of white cells, the ant oscillates infinitely many times, and then it visits the complete space, as we will see in next lemma.

Lemma 11. *An ant under the rule $\begin{pmatrix} 11 \\ 00 \end{pmatrix}$ that begins in a initial configuration with a finite number of white cells goes through all the cells of the space.*

Proof. Let us suppose that there exist cells that the ant does not visit; this implies (since the ant does not jump) that a complete hemisphere exists where the ant does not pass. Then, we can consider a cell α as the cell visited by the ant that is more to the right.

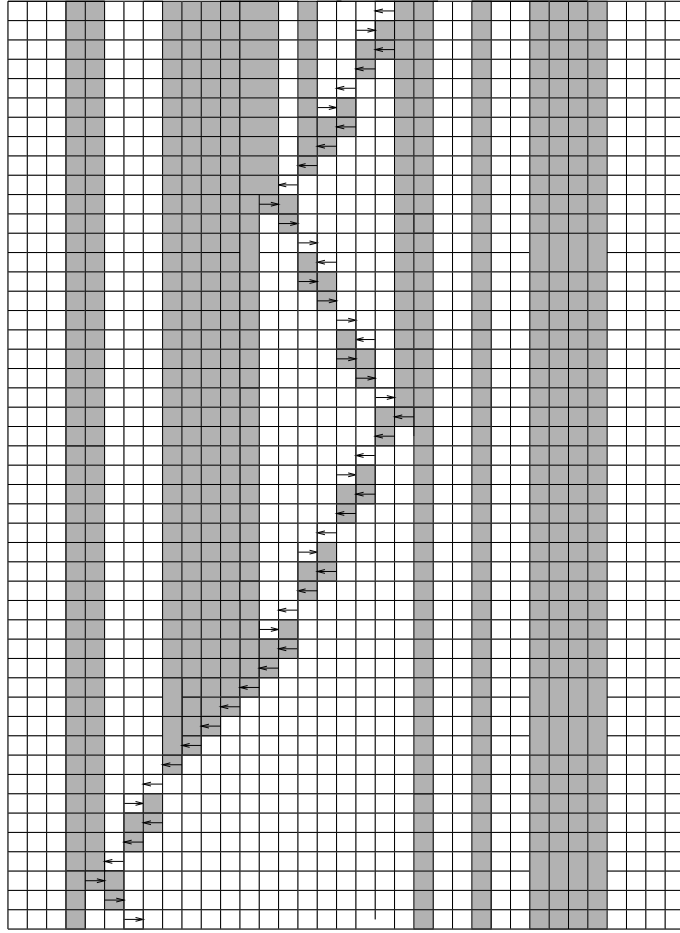


Figure 7.15: A simulation of the rule $\begin{pmatrix} 10 \\ 12 \end{pmatrix}$.

Figure 7.16 depicts the possibilities for the ant visiting α . In other cases the ant goes to the right of α immediately. Only the case in which the cell to the left of α is white is possible. The ant will advance to the left until falling in a black cell (remind that it always finds a black cell). It leaves all the visited cells in white color. When it finds a white cell it return arriving again to α . As α is white now, the ant continue to its right (Figure 7.17 shows this phenomenon). Then there are not such cell α and the ant must visit all the space. $\square$

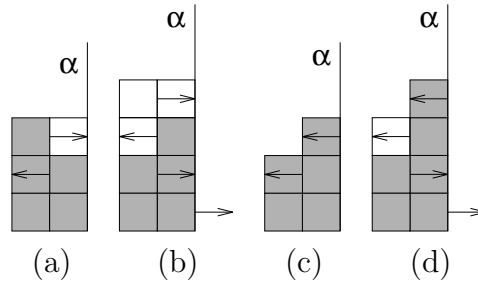


Figure 7.16: In cases (b) and (d) the ant goes to the right of α , but in cases (a) and (c) the ant remains to the left of α .

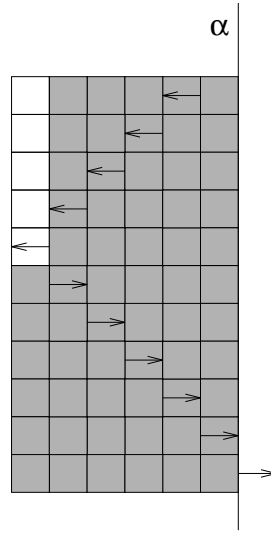


Figure 7.17: When the ant finds a white cell it returns to α .

7.2.2 The class U2

These rules differ from the previous ones because the ant eventually visits each cell of the space. Their behavior has “parabolic” form. This class is composed only by two rules.

Lemma 12. *An ant under the rule $\begin{pmatrix} 12 \\ 11 \end{pmatrix}$ or $\begin{pmatrix} 11 \\ 12 \end{pmatrix}$ that begins in a finite support initial configuration, goes by all the cells of the space.*

We will prove this Lemma by using similar arguments to those used in Lemma 11, that is to say that there exists just one pattern where the ant can propagate, and that when facing to a change, it rebounds but it enlarges the propagating pattern. As in this case the propagating pattern is more complex than in previous case, we need a preliminary Lemma.

From Lemma 3 we know that the ant with this rule advances at speed 1 through an alternating pattern of white and black cells. Since this ant always changes the color of the cells that it visits, when it passes through the sequence it yields a complementary alternating sequence.

Lemma 13. *Given a sequence of cells with alternating colors and an ant on the right of the sequence as follows: $\square \rightarrow$ or $\blacksquare \leftarrow$. If at the left of the sequence there are two cells with the same color. Then, the ant evolving under the rule $\begin{pmatrix} 12 \\ 11 \end{pmatrix}$, will advance to the last cell, and it will return to the starting point in the same direction of the beginning.*

Proof. Given a sequence of length n and an ant as in the hypothesis, let us suppose that at position $n + 1$ there is a cell that is in the same color that the n -th cell. From Lemma 3 the ant advances until the n -th cell, where the position of the ant satisfies the relation of the hypothesis.

Figure 7.18 shows what happens between the n -th cell and the $(n + 1)$ -th cell, when they are black and when they are white.

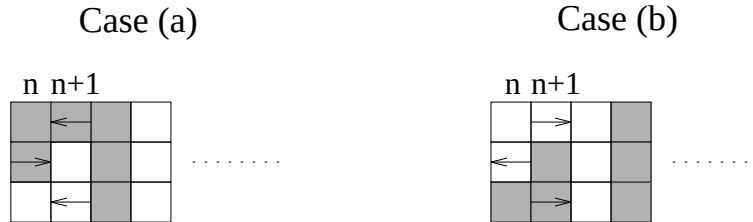


Figure 7.18: When the ant finds two cells with the same color, it returns.

In both cases the ant is to the left of an alternating sequence, in a position that is opposite to that of the hypothesis. Then, using Lemma 3, it advances to the right until arriving to their right side, in a position that is opposite to that of the hypothesis, because it is a displacement toward right. But, since the initial cell have now the opposed color, the ant points to the same direction that it had at the beginning. $\square$

Proof. (of Lemma 12)

We will prove the lemma for rule $\begin{pmatrix} 12 \\ 11 \end{pmatrix}$.

Let us suppose that there exist cells on which the ant does not pass; this implies (since the ant does not jump) that a complete hemisphere exists on which the ant does not pass. Then, we can consider a cell α as the cell visited by the ant that is rightmost.

When the ant visits α , and it is pointing toward right, it must be in white color, if not the ant moves to the right. If it is pointing toward left, α must be in black color, for the same reason. This color/position relation coincides with the hypothesis of Lemma 13.

Without lost of generality we can suppose the ant points toward left and α is black. Let us consider the sequence of cells that begins in α and it goes toward the left such that the color of the cells alternates.

It can have length one if the cell immediately to the left of α have the same color that α . Its length is finite since the number of black cells is finite.

From Lemma 13 we know the ant will return to α , finding it white and pointing to the left. Then the ant goes to the right, that contradicts the initial supposition.

□

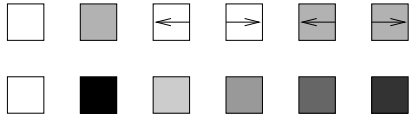
7.3 Conclusions

We have classified every one-dimensional ant's rule. In general one may say that the behavior is simple. For every rule we know its steady state and in this context, we have short-cut theorems depending only of the rule. Further, the steady state is also "simple". For bounded rules it is obviously periodic with short periods in the majority of cases (2 or 4). For the unbounded rules, essentially there are two steady behaviors: shift to the left or to the right, and constant and growing oscillations.

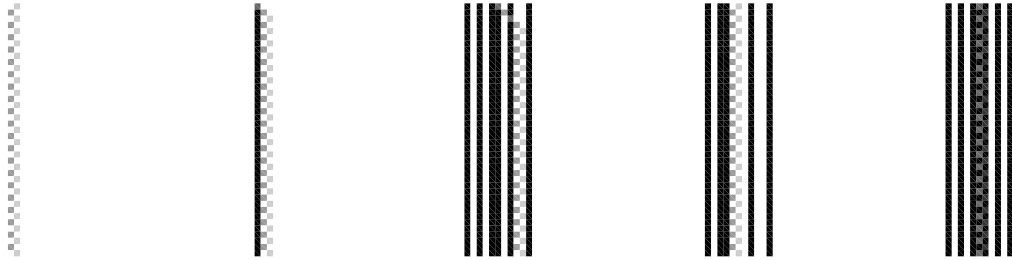
7.4 Simulations

Now we present some simulations of some representative rules of each class. A simulation was done over five different initial configurations. At the left, the configuration with all the cells in white state. After, just one black cell in the initial ant position.

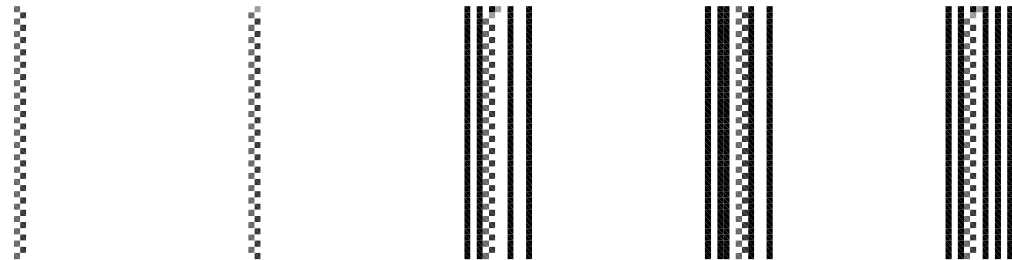
The presence of the ant over a cell is represented with gray tones. Different gray intensities represent different positions of the ant over different colors. The following diagram shows in the first row the representation used in previous figures, and in the second row the respective representation in the present figures:



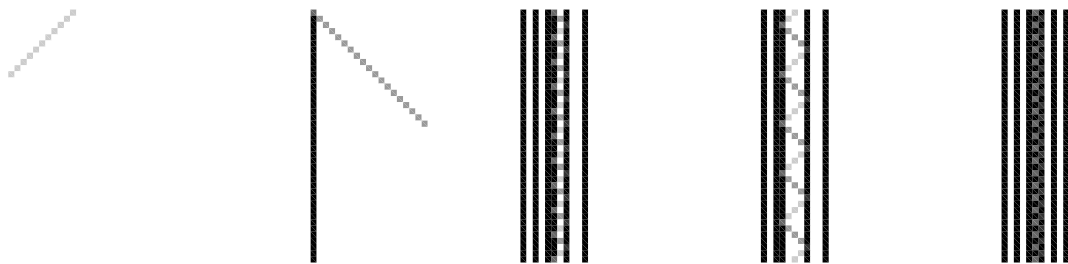
Class B1, rule $\begin{pmatrix} 01 \\ 02 \end{pmatrix}$:



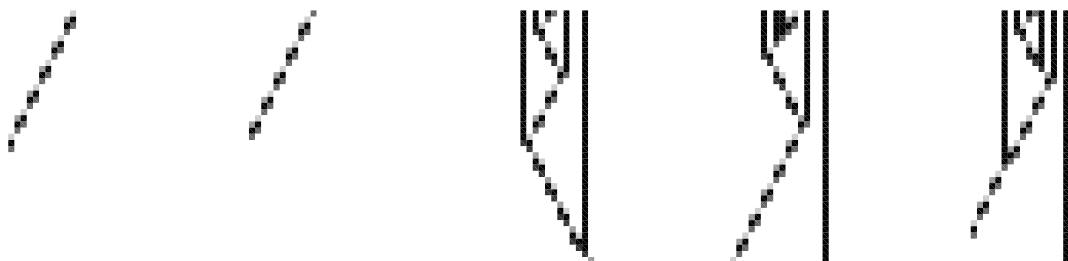
Class B1, rule $\begin{pmatrix} 10 \\ 01 \end{pmatrix}$:



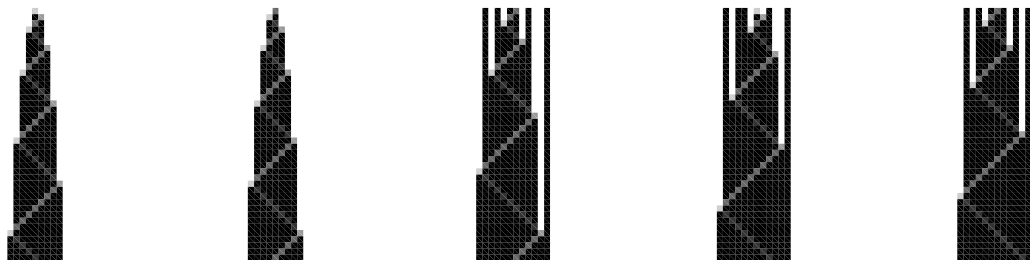
Class B2, rule $\begin{pmatrix} 01 \\ 00 \end{pmatrix}$:



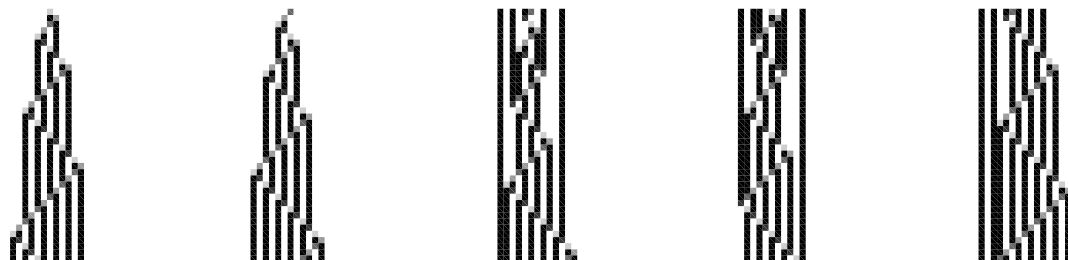
Class U1, rule $\begin{pmatrix} 10 \\ 12 \end{pmatrix}$:



Class U1, rule $\begin{pmatrix} 00 \\ 11 \end{pmatrix}$:



Class U2, rule $\begin{pmatrix} 12 \\ 11 \end{pmatrix}$:



Bibliography

- [1] E. R. Banks, “Information Processing and Transmission in cellular automata”, Ph.D. dissertation, Mass. Inst. of Tech., Cambridge, Mass., 1971.
- [2] E. Berlekamp, J. H. Conway and R. Guy, *Winning Ways for Your Mathematical Plays*, vol 2, (Academic Press., 1982).
- [3] O. Beuret, M. Tomassini, “Behaviour of Multiple Generalized Langton’s Ants”, *Artificial Life V*, pp 45-50, 1997.
- [4] L. A. Bunimovich, S. Troubetzkoy, “Recurrence properties of Lorentz Lattice Gas Cellular Automata”, *J. of Stat. Physics*, vol. 67, pp 289-302, 1992.
- [5] L. A. Bunimovich, S. Troubetzkoy, “Topological dynamics of flipping Lorentz lattice gas models”, *J. of Stat. Physics*, vol. 72, pp 297-307, 1993.
- [6] L. A. Bunimovich, S. Troubetzkoy, “Rotators, periodicity, and absence of diffusion in cyclic cellular automata”, *J. of Stat. Physics*, vol. 74, pp 1-10, 1994.
- [7] L. A. Bunimovich, “Many-dimensional Lorentz Cellular Automata and Turing Machines”, *Int. Jour. Bif. Chaos*, vol 6, pp 1127-1135, 1996.
- [8] A. W. Burks, *Essays on Cellular Automata*, Univ.Illinois Press, 1970
- [9] T. Chaboud, “Pavages et Graphes de Cayley Planaires”, Ph.D. dissertation, E. N. S., Lyon, 1995.
- [10] E. Cohen, New types of diffusion in lattice gas cellular automata, in: M. Mareschal, B. Holian, eds., *Microscopic Simulations of Complex Hydrodynamic Phenomena*, (Plenum Press, 1992).
- [11] J. H. Conway, “Unpredictable iterations”, in: *Proc. 1972 Number Theory Conference*, (University of Colorado, 1972).

- [12] J. H. Conway, J. C. Lagarias, "Tiling with polyominoes and combinatorial group theory", *Journal of Combinatorial Theory*, vol A53, pp 183-208, 1990.
- [13] R. Devaney, *Introduction to Chaotic Dynamical Systems*, (Addison Wesley Pub. Comp., 1989).
- [14] A. K. Dewdney, "Computer Recreations: Two-dimensional Turing Machines and Turmites Make Tracks on a Plane", *Scientific American*, September, pp 124-127, 1989.
- [15] A. K. Dewdney, "Computer Recreations", *Scientific American*, March, pp 118-121, 1990.
- [16] B. Durand, Zs. Roka, The game of life: universality revisited, in: M. Delorme, J. Mazoyer, eds., *Cellular Automata: a parallel model*, pp 51-76, (Kluwer Academic Pub., 1999).
- [17] A. Gajardo, E. Goles, "Universal cellular automaton over a hexagonal tiling with 3 states", to appear in *Int. J. Algebra and Computation*, 1998.
- [18] A. Gajardo, A. Moreira, E. Goles, "Complexity of Langton's ant", to appear in *Discret Applied Mathematics*, 2000.
- [19] A. Gajardo, A. Moreira, E. Goles, "Generalized Langton's Ant: Dynamical Behavior and Complexity", in: A. Ferreira, H. Reichel, eds., *STACS 2001, LNCS 2010*, (Springer-Verlag, 2001)
- [20] D. Gale, J. Propp, "Further Ant-ics", *Math. Intelligencer*, vol. 16, N. 1, pp 37-42, 1994.
- [21] D. Gale, J. Propp, S. Sutherland, S. Troubetzkoy, "Further Travels with My Ant", <http://www-math.mit.edu/~propp/ant.ps.gz> and *Math. Intelligencer*, vol. 17, N. 3, pp 48-56, 1995.
- [22] D. Gale, *Tracking the Automatic ANT and Other Mathematical Explorations*, Springer-Verlag, N. Y., 1998.
- [23] E. Goles, A. Maass, S. Martínez, "On the limit set of some universal cellular automata", *Teoret. Comput. Sci.*, vol 110, pp 53-78, 1993.
- [24] E. Goles, M. Margenstern, "Sand pile as a universal computer", *Int. J. Modern Physics C*, vol. 7, N. 2, pp 113-122, 1996.

- [25] P. Groszfil, J. P. Boon, E. G. D. Cohen, L. A. Bunimovich, “Propagation and organization in a lattice random media”, *J. Stat. Physics*, vol 97, N. 3-4, pp 575-608, 1999.
- [26] B. Kitchens, *Symbolic Dynamics*, (Springer, 1998).
- [27] X. P. Kong and E. G. D. Cohen, “Diffusion and propagation in a triangular Lorentz lattice gas cellular automata”, *J. Stat. Physics*, vol 62, pp. 737-757, 1991.
- [28] P. Kůrka, “On topological dynamics of Turing machines”, *Teoret. Comput. Sci.*, vol 174, N 1-2, pp 203-216, 1997.
- [29] C. G. Langton, “Studing artificial life with cellular automata”, *Physica D*, vol 22D, pp 120-149, 1986.
- [30] D. Lind, B. Marcus, *Symbolic Dynamics and Coding*, (Cambridge Univ. Press, 1995).
- [31] K. Lindgren, M. G. Nordahl, “Universal Computation in Simple One-Dimensional Cellular Automata”, *Complex Systems*, 4, pp. 299–318, 1990.
- [32] A. Maass, “Universal cellular automaton simulating any programable machine”, *Revista de Matematicas Aplicadas*, vol. 12, pp 107-127, 1991.
- [33] H. Meng, E. G. D. Cohen, “Growth, self-randomization, and propagation in a Lorentz lattice gas”, *Physical Rev. E*, vol 50, N. 4, pp 2482-2487, 1994.
- [34] M. L. Minsky, *Computation: Finite and infinite machines*, Englewood Cliffs, N.J., Prentice–Hall, 1967.
- [35] N. Margolus, “Physics-like models of computations,” *Physica D*, vol 10, pp. 81-95, 1984.
- [36] C. Papadimitriou, *Computational Complexity* (Addison-Wesley, 1994).
- [37] C. Papazian, E. Rémila, “Some properties of hyperbolic discrete planes”, Accepted for DGCI 2000
- [38] C. Papazian, personal communication.
- [39] C. Pickover, “Picturing randomness with Truchet tiles”, *J. Recreational Math.*, vol 21, pp. 256, 1989.
- [40] J. Propp’s personal web page, <http://kleene.math.wisc.edu/~propp/>

- [41] E. Rémila, personal communication.
- [42] M. Sipser, *Introduction to the Theory of Computation* (PWS Publishing Company, 1996) .
- [43] A. R. Smith III, “Simple Computation-Universal Cellular Spaces.” *Journal of the ACM.*, vol 18, N 3, pp.339-356, 1971.
- [44] S. Troubetzkoy, “Lewis–Parker Lecture 1997 The Ant” , *Alabama Journal of Mathematics.*, vol 21(2), (1997).
- [45] J. de Vries, *Elements of Topological Dynamics*, (Kluwer Academic Pub., 1993).
- [46] F. Wang, “Motion in A Lorentz Lattice Gas” , Ph. D. dissertation, The Rockefeller University, , 1995.