

III Escuela de Verano, 9 al 21 de Enero de 2006
Departamento de Ingeniería Matemática
Universidad de Concepción

Redes Dinámicas Discretas

Julio Aracena y Anahí Gajardo

Índice de contenidos

1. Definiciones elementales

1.1. Grafos

Grafos dirigidos y no dirigidos. Notación, vecindad, grado, regularidad. Recorridos, caminos, circuitos, ciclos, conexidad, distancia, diámetro. Árboles. Conjuntos independientes y bipartición. Planaridad.

1.2. Sistema dinámico a tiempo discreto

Órbita. Inyectividad, sobreyectividad y reversibilidad. Punto fijo, punto periódico, conjunto de puntos periódicos, puntos últimamente periódicos y transiente. Atractor y base de atracción. Grafo de transición.

1.3. Redes dinámicas discretas

2. Herramientas de estudio

2.1. Funcionales de Lyapunov

2.2. Complejidad algorítmica

3. Hormiga de Langton en un grafo finito

4. Redes Booleanas

5. Redes neuronales discretas

1. Definiciones elementales

1.1. Grafos

Existen fundamentalmente dos tipos de grafos, los grafos dirigidos (o digrafos) y los no dirigidos.

Un *grafo dirigido* es un par (V, E) , donde V es un conjunto llamado conjunto de *vértices* (o *nodos*), y $E \subseteq V \times V$

Un *grafo no dirigido* es un par (V, E) , donde V es un conjunto, y $E \subseteq \mathcal{P}(V)$ es un conjunto cuyos elementos son conjuntos de cardinalidad exactamente igual a 2, es decir, es un conjunto de pares **no ordenados** de vértices.

En ambos casos, E es llamado conjunto de *aristas* (o *arcos*).

Si bien la mayoría de las nociones siguientes son válidas cuando V es infinito, a partir de ahora nosotros supondremos que V es un conjunto finito.

Si $(u, v) \in E$ (o $\{u, v\} \in E$) decimos que u y v son *vecinos* o *adyacentes*; decimos además que u *incide a* v . Se dice que la arista (u, v) (o $\{u, v\}$) *une a* u con v y que u y v son sus *extremos*. Si un vértice u aparece en dos aristas, se dice que éstas son *adyacentes*. Una arista del tipo (u, u) se llama *bucle* o *loop*. Un grafo no dirigido no puede tener bucles.

La *vecindad* de un vértice v en el caso de los digrafos, se descompone en dos conjuntos, el *conjunto de sucesores* $N_+(v) = \{u \in V : (v, u) \in E\}$ y el *conjunto de predecesores* $N_-(v) = \{u \in V : (u, v) \in E\}$. Así, la vecindad de v es: $N(v) = N_+(v) \cup N_-(v)$. En el caso no dirigido, la vecindad es simplemente: $N(v) = \{u \in V : \{u, v\} \in E\}$.

Si un vértice no tiene vecinos, decimos que es un *vértice aislado*.

El *grado* de un vértice es el tamaño de su vecindad. Así denotamos: $d_+(v) = |N_+(v)|$, $d_-(v) = |N_-(v)|$ y $d(v) = |N(v)|$.

Un grafo se dice *regular* de grado k o *k-regular* si todos sus vértices tienen el mismo grado, es decir, si existe $k \in \mathbb{N}$ tal que para todo $v \in V$, $d(v) = k$.

Un *recorrido* es una secuencia ordenada de vértices v_0, \dots, v_n tales que $(v_i, v_{i+1}) \in E$ (o $\{v_i, v_{i+1}\} \in E$) para todo $i \in \{0, \dots, n-1\}$. Un *circuito* es un recorrido v_0, \dots, v_n tal que $v_0 = v_n$. El *largo* de un recorrido v_0, \dots, v_n es n .

Un *camino* $P = v_0, \dots, v_n$ se define como un recorrido *elemental*, es decir, un recorrido que no repite ningún vértice, a excepción eventualmente de los extremos. Un *ciclo* $C =$

v_0, \dots, v_n es un camino cerrado, es decir donde $v_0 = v_n$.¹ si no repite ningún vértice, es decir, si v_0, \dots, v_{n-1} es un recorrido elemental.

Se dice que u está *conectado* con v si existe un camino que parte en u y termina en v .

Observación 1 *Si existe un recorrido de u a v , entonces también existe un camino de u a v .*

Se dice que un grafo es *fuertemente conexo* si para todo $u, v \in V$, u está conectado con v .

La distancia de u a v es el largo del camino más corto que va de u a v , y se denota por $d(u, v)$. Si u no está conectado con v , entonces $d(u, v) = \infty$. Observamos que en un grafo no dirigido $d(u, v) = d(v, u)$, lo cual no necesariamente se cumple en un grafo dirigido. El diámetro de un grafo es la máxima distancia entre sus vértices: $\max\{d(u, v); u, v \in V\}$.

Cuando el grafo es no dirigido y tiene al menos una arista, siempre tendrá al menos un circuito de largo 2; Ej.: u, v, u es un circuito si $\{u, v\} \in E$. Si el grafo es conexo y no tiene ciclos decimos que es un *árbol*.

Observación 2 *En un árbol existe un único camino entre cada par de vértices*

Una *representación plana* de un grafo es un dibujo de éste en el plano, en que los vértices se representan por puntos y los arcos por líneas que unen sus extremos sin pasar por ningún otro vértice. Un grafo se dice *planar* si tiene una representación plana en que las líneas que representan sus arcos no se cruzan.

Por último, dado un grafo (dirigido o no) $G = (V, A)$, $S \subset V$ se dice *independiente (IS)* si para todo par de vértices de S no existe un arco que los conecte. $S \subset V$ se dirá *independiente maximal (MIS)* si es IS y no está contenido en otro conjunto IS. Un grafo se dice *bipartito* si el conjunto de nodos puede ser particionado en dos subconjuntos IS.

¹ No hay acuerdo en la literatura respecto a la denominación de estos conceptos. Muchos textos prefieren la expresión “recorrido elemental” en lugar de “camino” y “circuito elemental” en lugar de “ciclo”. También existe la noción de “recorrido simple” que se refiere a un recorrido en que no se repiten aristas.

1.2. Sistema dinámico a tiempo discreto

Un *sistema dinámico a tiempo discreto* es una dupla (X, T) donde X es un conjunto llamado conjunto de *estados* (o espacio de fase) y $T : X \rightarrow X$ es una función llamada *función de transición*. Un elemento x de X se llama *estado* del sistema. Si x es el estado inicial del sistema, $T^t(x)$ es el estado en el tiempo t .

La *órbita* de un punto $x \in X$ es el conjunto de puntos: $\{T^n(x)\}_{n \in \mathbb{N}}$ (con \mathbb{N} incluyendo el 0).

Un sistema dinámico se dice *inyectivo* (*sobreyectivo*) si la función T es inyectiva (sobreyectiva). Si T es biyectivo, se dice que el sistema es *reversible*.

Cuando un sistema no es sobreyectivo existe un conjunto de puntos que no tienen preimagen; este conjunto se conoce con el nombre de *jardín del edén*.

Observación 3 Si X es finito, entonces la inyectividad resulta equivalente a la sobreyectividad, de manera que en este caso todos los sistemas inyectivos serán reversibles.

Un punto x es un *punto fijo* del sistema si satisface $T(x) = x$. Observamos que en este caso su órbita tiene un sólo punto: $\{T^n(x)\}_{n \in \mathbb{N}} = \{x\}$.

Un punto x es *periódico* si existe $p \in \mathbb{N} - \{0\}$ tal que $T^p(x) = x$; si p es el natural más pequeño que satisface esto se dice que p es el *periodo* de x . Observamos que en este caso la órbita de x tiene p elementos: $\{T^n(x)\}_{n \in \mathbb{N}} = \{T^n(x)\}_{n=0}^{p-1}$ y se denomina ciclo límite o ciclo dinámico. Si $p = 1$, se trata de un punto fijo.

El conjunto de puntos periódicos de un sistema (X, T) se denota por $P(X, T)$.

Un punto x se dice *últimamente periódico* si existe un natural $t \in \mathbb{N} - \{0\}$ tal que $T^t(x)$ es periódico; si t es el natural más pequeño que satisface esto se dice que t es el *transiente* de x . Observamos que si x tiene transiente t y periodo p entonces su órbita tiene $t + p$ elementos. Recíprocamente, si la órbita de un punto x es finita, significa que existe t tal que $T^t(x) = T^s(x)$ para algún $s < t$; lo cual significa que x es últimamente periódico.

Observación 4 Si X es finito, todas las órbitas son finitas, luego todos los puntos son periódicos o últimamente periódicos.

Un conjunto $A \subseteq X$ es un *atractor* si $T(A) = A$. Es decir, los atractores pueden ser puntos fijos o ciclos límites. Si además no contiene estrictamente a ningún atractor, decimos que es un *atractor minimal*.

Dado un atractor A , se define su *valle de atracción* como:

$$V(A) = \{x \in X : (\exists t \in \mathbb{N}) T^t(x) \in A\}$$

que consiste en todos los puntos que “caen” en A en un tiempo finito. Si X fuese un espacio topológico se podría definir el valle de atracción como los puntos que “convergen” a A ; pero en este curso trabajaremos sólo con espacios finitos, donde la única topología interesante da lugar a la definición anterior.

- Observación 5**
1. *Cuando X es finito, los atractores minimales consisten simplemente en la órbita de algún punto periódico. Así, el conjunto de atractores de un sistema dinámico finito corresponde a su conjunto de puntos periódicos.*
 2. *Puesto que todos los puntos son últimamente periódicos, todos pertenecen a algún valle de atracción; así, la familia de valles de atracción particiona el espacio.*

Una manera de visualizar un sistema dinámico a tiempo discreto en un espacio finito es dibujar su grafo de transición. El *grafo de transición* asociado a (X, T) es el grafo dirigido $G = (X, E)$, donde E está definido por $E = \{(x, T(x)) : x \in X\}$.

- Observación 6**
1. *Cada nodo x tiene un único vértice de salida: $T(x)$.*
 2. *Si T es inyectivo, cada nodo tiene a lo más un vértice de entrada.*
 3. *Si T es sobreyectivo, cada nodo tiene al menos un vértice de entrada.*
 4. *Debido a lo anterior, el grafo de un sistema cualquiera siempre consistirá en uno o más anillos (atractores periódicos) de donde cuelgan árboles.*
 5. *Si es reversible, sólo consistirá en un conjunto de anillos (y/o bucles).*
 6. *Interesa conocer: el número de atractores minimales, su tamaño, el transiente máximo, el tamaño de los valles de atracción.*

1.3. Redes dinámicas discretas

Una *red dinámica discreta* $N = (G = (V, A), Q, F)$ está definida por una tupla donde G es un grafo dirigido o no (finito o infinito), Q es un conjunto finito llamado conjunto de estados y $F = \{f_i\}_{i \in I}$ es una familia de funciones de activación local. De esta forma, N es un tipo particular de sistema dinámico (X, F) donde $X = Q^{|V|}$ y $F(x)_i = f_i(x_j : j \in N_-(i)), \forall x \in X, i \in V$.

2. Herramientas de estudio

Cuando X es finito, todos los puntos son últimamente periódicos. El periodo de un punto es a lo más igual a la cardinalidad del conjunto. A veces ésta puede ser bastante grande, entonces nos interesa saber si existe una cota más estrecha. En la Sección 2.1 mostramos la técnica más usada para demostrar que el periodo de los puntos o equivalentemente el largo de los ciclos límites de un sistema tiene una cota “pequeña”.

En general calcular los valles de atracción de un sistema es un problema “difícil”. En la Sección 2.2 explicamos cómo se mide la dificultad de un problema.

2.1. Funcionales de Lyapunov

Un sistema dinámico (X, T) tiene un funcional de Lyapunov, si existe una función $\xi : X \rightarrow \mathbb{R}$ tal que para todo $x \in X$, $\xi(x) > \xi(T(x))$ salvo que $T(x) = x$, tal función se llama *funcional de Lyapunov*.

Teorema 1 *Si (X, T) tiene un funcional de Lyapunov y X es un conjunto finito, entonces (X, T) sólo tiene puntos que convergen a puntos fijos.*

El siguiente corolario permite usar esta herramienta para en sistemas con puntos periódicos.

Corolario 1 *Si (X, T^p) tiene un funcional de Lyapunov y X es un conjunto finito, entonces (X, T) sólo tiene puntos últimamente periódicos de periodo menor o igual a p .*

2.2. Complejidad algorítmica

Frecuentemente uno estudia un sistema dinámico con el fin de poder *predecirlo*. Saber, por ejemplo, si un punto y pertenece o no a la órbita de un punto x . O si en tiempo estacionario (una vez que ha entrado en un atractor), los puntos visitados serán de tal tipo o de otro.

Para responder tales preguntas basta con calcular la órbita del punto que nos interesa, hasta verificar la propiedad que uno busca. Sin embargo, este proceso puede tomar mucho *tiempo*; la órbita puede pasar por casi todos los puntos del espacio antes de verificar la propiedad que buscamos, y todos los puntos del espacio, incluso tratándose de un espacio finito, puede ser un gran número. Entonces esta estrategia no es buena.

Mejor sería poder caracterizar matemáticamente los puntos que satisfacen la propiedad buscada, de manera que evaluar tal característica sea *simple*, es decir, que haya un

método que nos permita hacerlo en *poco tiempo*.

Lamentablemente la Informática Teórica ha demostrado que existe una jerarquía de complejidad que establece que a cada problema está asociado un *nivel de dificultad intrínseco* que no es posible superar. Veamos cómo funciona esto.

2.2.1. Problema

La teoría afecta solamente los llamados problemas de *decisión*. Pero esto no es una restricción muy importante pues todos los problemas pueden escribirse como problemas de decisión. Informalmente un problema de decisión consiste en determinar, si un objeto dado pertenece o no a una clase dada. Formalmente entonces, un problema de decisión está definido por el conjunto de objetos I sobre el cual trata, llamado conjunto de *instancias* y la clase por la cual se desea interrogar I_+ , llamada conjunto de *instancias positivas*.

Un problema es entonces un par ordenado de conjuntos (I, I_+) tales que $I_+ \subseteq I$.

2.2.2. Complejidad de un algoritmo

Un algoritmo es una secuencia definida y finita de instrucciones a realizar a sobre un objeto o entrada finita. Para cada entrada x , $f(x)$ se define como el *tiempo* que demora el algoritmo en terminar, estando dado éste por el número de operaciones realizadas. Suponiendo además que se dispone de una medida del *tamaño* de las entradas $|x|$, se estudiará la función $g(n) = \max\{f(x) : |x| = n\}$ que indica el tiempo máximo tomado por el algoritmo en las entradas de tamaño n .

El tipo de crecimiento que tiene g define la *complejidad algorítmica* del algoritmo.

Dada una función $h : \mathbb{N} \rightarrow \mathbb{N}$ se dice que el algoritmo es *de orden h* , y se escribe $g(n) = O(h(n))$ si existen $n_0, c \in \mathbb{N}$ tales que para todo $n > n_0$, $g(n) \leq c h(n)$.

Se dice que el algoritmo es *polinomial* si h es una función polinomial, se dice que es *exponencial* si h es una función exponencial, etc.

2.2.3. Complejidad de un problema

Se dice que un algoritmo *resuelve* el problema (I, I_+) si para cada entrada $x \in I$ el algoritmo se detiene en una cantidad finita de pasos, entregando como salida la respuesta “Sí” cada vez que $x \in I_+$ y la respuesta “No” cuando $x \notin I_+$.

Diremos, siendo un poco informales, que un problema está en una determinada clase de complejidad (polinomial, exponencial, etc.) si existe un algoritmo que lo resuelve y que tiene la complejidad indicada.

Esta definición es perfectamente justa, pero podría ser completamente inútil si para cada problema existiera siempre un algoritmo tan simple como se quiera. Lamentablemente esto no es así, como lo justifica el siguiente teorema, que es tal vez el más importante del área. Éste indica que existe un problema exponencial, para el cual no existe ningún algoritmo polinomial que lo resuelva.

Teorema 2 *La clase de problemas polinomiales está estrictamente contenida en la clase de problemas exponenciales.*

3. Hormiga de Langton en un grafo finito

En las últimas décadas, se han introducido diversos sistemas dinámicos discretos simples (pilas de arena, autómatas celulares, etc.). Ellos son de interés en matemática, 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 la física y la biología.

La hormiga de Langton aparece por primera vez en la década del 80 [9] como un modelo de vida artificial. Fue definida originalmente en la grilla cuadrada infinita. Su comportamiento en ese caso se muestra complicado y aún sin explicación. Tal como muestra la Figura 1, al partir con sólo algunas células en estado negro, termina siempre cayendo en un movimiento periódico que deja una traza regular característica. 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”.

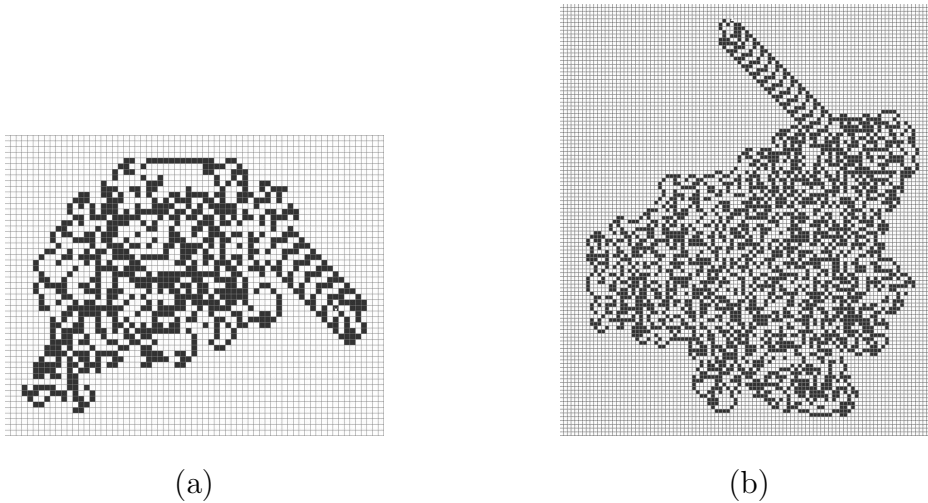


Figura 1. (a) Configuración en la iteración 10700, cuando parte con todas las células en estado blanco. (b) Configuración en la iteración 47500, a partir de una configuración con una célula en estado negro.

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 [3] y de las “lombrices programadas” [4,5].

Su definición informal es muy natural: una hormiga se desplaza por los arcos de un grafo dibujado en el plano cuyos vértices 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 está de color negro (respectivamente blanco). Además, al pasar por cada vértice, hace que éste cambie de color. La Figura 2 muestra un ejemplo de un estado del sistema. Veamos ahora la definición formal.

Definición 1 Dada la representación plana de un grafo no dirigido (V, E) . Si en una iteración dada, la hormiga está en la posición $h_t = (i, j)$, sobre el arco $\{i, j\}$, apuntando al vértice j , y si los estados de cada vértice 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 los vértices estarán dadas por:

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

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

$$c_{t+1}(k) = c_t(k) \text{ si } k \neq j$$

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.

El conjunto de estados es entonces: $X = \{\text{blanco}, \text{negro}\}^V \times A$, donde $A = \{(i, j) : \{i, j\} \in E\}$. Aquí $\{\text{blanco}, \text{negro}\}^V$ representa el conjunto de las configuraciones, es decir de las funciones que a cada vértice le asignan un color, blanco o negro. La función de transición $T : X \rightarrow X$ queda definida por $T(c_t, h_t) = (c_{t+1}, h_{t+1})$.

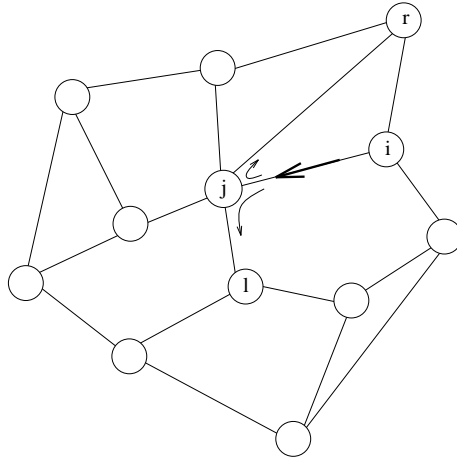


Figura 2. La hormiga es un par ordenado de vértices adyacentes (i, j) . Al llegar a cada vértice, ella dobla lo más posible a la derecha o a la izquierda.

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. Supondremos que (V, E) es conexo, pues si no, la hormiga no tendrá acceso a todos los vértices.

No es difícil ver que el sistema es reversible. Basta observar que el vértice al cual la hormiga apuntaba en la iteración anterior era el que ahora es su “cola”. Por lo tanto, a partir del color de este vértice es posible deducir en qué dirección dobló, y de allí deducir en qué posición estaba antes. Los resultados que se presentan en esta sección se encuentran en [7].

Proposición 1 *La función T de la Definición 1 es biyectiva.*

Observación 7 1. *Si la hormiga llega a un nodo de grado 1, simplemente “da media vuelta”, independientemente del estado de éste.*

2. *Si llega a un nodo de grado 2, de manera independiente de su estado, sale por el otro arco adyacente al nodo en cuestión.*

3.1. Estudio del tamaño de los periodos

Como el sistema es reversible, sólo tiene puntos periódicos. El periodo máximo de un sistema está siempre acotado por la cardinalidad de X ; en este caso es $2^{|V|+1}|E|$. Pero ¿de qué tamaño son realmente? son similares a $2^{|V|+1}|E|$ o son mucho más cortos. Seguramente esto depende del grafo. Partamos estudiándolo para el caso más simple: un árbol.

Proposición 2 *Si el grafo es un árbol, el periodo de todo puntos está acotado por 4 veces el diámetro del grafo.*

Ejercicio. *¿Cuántos atractores minimales tiene el sistema para el grafo de la Figura 3? Dé una cota inferior para el número de atractores minimales en un árbol cualquiera.*

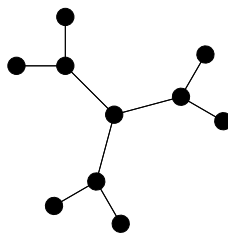
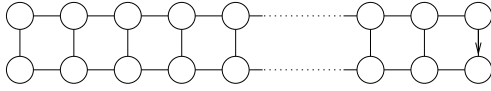


Figura 3. Un árbol binario de diámetro 4.

Si el grafo no es un árbol inmediatamente aparecen ciclos de largo exponencial. A continuación mostramos una familia de grafos que poseen puntos periódicos de periodo exponencial en el número de vértices, así como puntos de periodo lineal. Mostrando que un mismo grafo puede dar lugar a comportamientos diversos. La Figura 4 muestra estos grafos.

Proposición 3 *Si el grafo es una grilla cuadrada de alto dos y ancho n , entonces existe un punto cuyo periodo es 2^n y otro cuyo periodo es $4(n - 1)$.*

Periodo exponencial



Periodo lineal

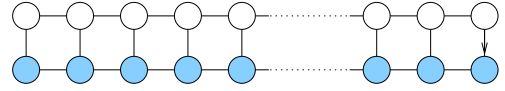


Figura 4. Grilla cuadrada de altura dos y ancho arbitrario. Configuraciones que definen puntos con distinto periodo.

Está abierto lo que ocurre en grafos arbitrarios. ¿Depende el periodo del “número de circuitos” si éste se puede definir de alguna forma? Un caso particularmente patológico es el de la grilla hexagonal donde, a pesar de estar la teoría bastante desarrollada, no ha sido posible calcular ni siquiera el periodo del punto más sencillo: cuando todos los vértices tienen el mismo color.

3.2. Estudio de la predecibilidad del sistema

Demostraremos que predecir este sistema es al menos tan difícil como calcular una fórmula lógica. Esto lo haremos mostrando que es posible hacer que el sistema calcule, a través de su dinámica, el valor de cualquier fórmula lógica dada. El problema de decisión que estudiaremos es el siguiente:

- (H) *Una instancia está dada por un grafo no dirigido, una asignación de estados a sus vértices, una posición de la hormiga y la indicación de un vértice particular del grafo.*

Una instancia es positiva si, en el grafo, partiendo de la asignación de estados y la posición dada, la hormiga visita en algún momento el vértice indicado.

Formalmente lo que haremos es “reducir” el problema de decisión CIRCUITVALUE a (H).

CIRCUITVALUE *Una instancia está dada por una fórmula lógica y una asignación de valores a sus variables.*

Una instancia es positiva si la fórmula lógica toma valor verdadero sobre la asignación dada.

Para reducir el problema CIRCUITVALUE a (H) debemos, asociar, a cada fórmula lógica una instancia de (H), de manera que la fórmula lógica se satisfaga si y sólo si la hormiga visita el vértice dado.

Esto se lleva a cabo haciendo que, de alguna manera, la hormiga calcule la fórmula lógica. Veremos que esto es posible. Para comenzar, logramos que la hormiga calcule fórmulas atómicas; la Figura 5 muestra cómo calcular una negación y una función “o”. Interpretamos el color blanco como el valor lógico *falso* y el negro como *verdadero*.

Inicialmente se asigna al vértice a un color representando un valor lógico, y al vértice s el color blanco.

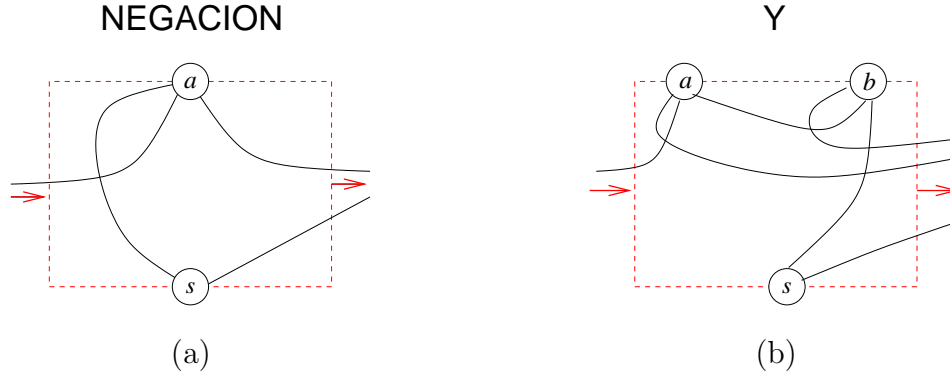


Figura 5. (a) Al entrar la hormiga a esta parte del grafo desde la izquierda, se cumplirá que visitará el vértice de s si y solamente si el vértice a está en estado blanco. Además, si hacemos que s siempre parta en color blanco, su color final será el opuesto al color inicial de a . (b) Si pasa por ésta parte del grafo visitará s si y sólo si a o b están en estado negro, y sólo en tal caso el estado de s será negro al salir la hormiga.

Veamos ahora cómo calcular una fórmula completa. Primero recordamos que cualquier fórmula lógica se puede escribir usando sólo funciones “y” binarias y negaciones. Entonces es posible calcularla iterativamente, partiendo con las expresiones atómicas, luego calculando las que ligan a estas y así sucesivamente.

Para hacer ésto con la hormiga es necesario poder:

- 1) recuperar el cálculo de las fórmulas atómicas;
- 2) duplicar este valor para usarlo todas las veces que sea necesario;
- 3) desplazarlo hacia la posición deseada.

La idea es encadenar los bloques que calculan las fórmulas atómicas, para así calcular fórmulas complejas. Esto es posible gracias a que el vértice s registra el valor de verdad de la fórmula donde está. Así, se puede transformar en el vértice de entrada de otro bloque puesto abajo. Esto permite encadenar las fórmulas en forma vertical. También es posible encadenar las fórmulas en forma horizontal, haciendo que al salir de un bloque, la hormiga entre en otro bloque. Como de cada bloque salen varios arcos, debemos agregar un trozo de grafo que haga haga las veces de “unión” de estos arcos. La Figura 6 muestra un bloque que logra ésto.

Las condiciones 2) y 3) se logran gracias a los bloques CRUCE y DUPLICADOR que se muestran en la Figura 7. Encadenando las fórmulas y haciendo pasar la hormiga por cada una, como lo muestra la Figura 8 (que lo hace para el caso de la función “o excluyente”) es posible calcular fórmulas complejas.

De esta manera probamos que, en general, predecir el comportamiento de un sistema es al menos tan difícil como calcular una fórmula lógica, lo cual es polinomial. Esto hace pensar que no es posible predecir un sistema en tiempo logarítmico, por ejemplo.

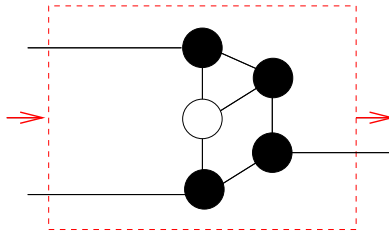


Figura 6. Si la hormiga entra desde la izquierda, por el arco de abajo, saldrá inmediatamente por la derecha. Si entra por el arco de arriba, dará algunas vueltas, pero finalmente saldrá también por la derecha.

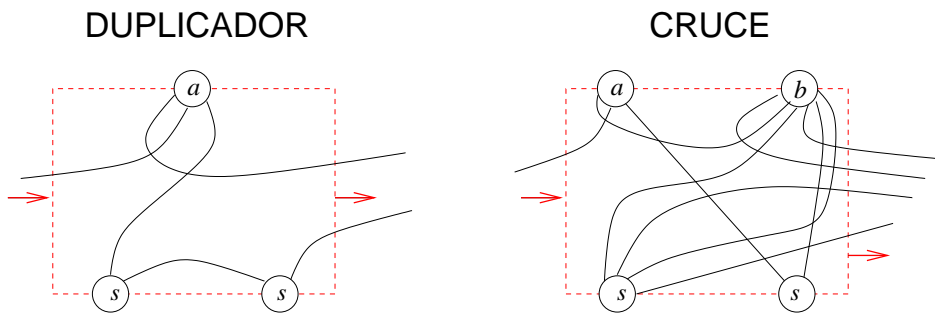


Figura 7. Estos bloques permiten cruzar y duplicar información.

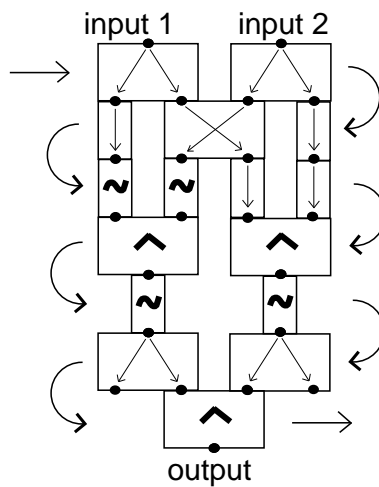


Figura 8. Haciendo pasar sucesivamente la hormiga por los diferentes bloques es posible calcular la función “o excluyente”. La hormiga pasará por el último vértice solamente si el color de los primeros vértices satisfacen el “o excluyente”.

También muestra que este tipo de sistemas pueden ser usados para calcular.

4. Redes Booleanas

Las redes Booleanas fueron introducidas por Stuart Kauffman en el año 1969 [11] para modelar redes de interacción génica. Formalmente las redes Booleanas son redes dinámicas discretas donde el conjunto de estados es binario, normalmente, $Q = \{0, 1\}$.

4.1. Redes Booleanas Aleatorias

Desde la introducción de las redes Booleanas existen diversas investigaciones respecto al comportamiento dinámico de estas redes. El siguiente cuadro resume los comportamientos dinámicos observados en redes Booleanas aleatorias donde el grafo de interacción es de tipo regular.

κ	Largo de los ciclos	Número de ciclos	estabilidad
1	$\exp(\frac{1}{8} \log^2 N)$	$\frac{1}{\sqrt{e}} N(1 + O(1))$	baja
2	\sqrt{N}	\sqrt{N}	alta
> 3	$2^{\beta N-1}, \beta$	$\frac{1}{2} N$	baja
N	$2^{\frac{N}{2}-1}$	$\frac{N}{e}$	baja

Como se puede observar del cuadro resumen las redes booleanas aleatorias experimentan una diferencia de comportamiento dinámico importante cuando la conectividad del grafo cambia de $k = 2$ a $k = 3$. Se pasa de un comportamiento más bien ordenado con largo de transientes y ciclos no muy grandes a uno más caótico donde se pueden observar ciclos de largo exponencial. Este cambio de conectividad y cambio de comportamiento dinámico es lo que se conoce como frontera del caos en redes Booleanas.

4.2. Redes AND-OR.

Pese a la dificultad de conocer el comportamiento dinámico de una red discreta general a partir sólo de su definición formal, por lo sensible de este frente a pequeños cambios de la red, es posible probar ciertos resultados generales para algunas familias particulares de redes Booleanas, por ejemplo las redes AND-OR.

Las redes AND-OR son un caso particular de redes Booleanas, esto es, $Q = \{0, 1\}$ y cuyas funciones de activación local son del tipo AND u OR. Es decir, $\forall x = (x_1, \dots, x_n) \in Q^n$, $f_i(x) = AND(x)$ significa que $f_i(x) = 1$ si y sólo si $\forall j, x_j = 1$ y $f_i(x) = OR(x)$ equivale a decir que $f_i(x) = 1$ si y sólo si $\exists j, x_j = 1$. Forlamente una red AND-OR N puede ser definido por una tripleta $N = (G, V_{AND}, V_{OR})$, donde $G = (V, E)$ es un

digrafo finito, $V_{AND} \subset V$ y $V_{OR} \subset V$ son subconjuntos de vértices donde $\forall i \in V_{AND}$ f_i es de tipo *AND* y $\forall j \in V_{OR}$, f_j es de tipo *OR*.

4.3. Máximo número de puntos fijos de redes AND-OR.

En esta sección estudiaremos el máximo número de puntos fijos de redes AND-OR, con respecto al tamaño del conjunto de nodos. Las demostraciones de esta sección y resultados complementarios pueden ser encontrados en [1]

Es fácil verificar que en el caso de N una red AND-OR donde todas las funciones son del tipo OR o donde todas las funciones de tipo AND los únicos puntos fijos son los vectores $\vec{0} = (0, 0, \dots, 0)$ y $\vec{1} = (1, 1, \dots, 1)$ (ejercicio). De igual forma probaremos más adelante que los ciclos dinámicos en redes AND-OR son todos de largo 2.

Sin embargo, probaremos que las redes AND-OR en general pueden llegar a tener un número exponencial de puntos fijos. Previamente el siguiente teorema.

Teorema 3 *Sea $N = (G, V_{AND}, V_{OR})$ una red AND-OR con G digrafo conexo sin loops y S su conjunto de puntos fijos. Entonces, existe una red AND-OR $\widehat{N} = (\widehat{G}, \widehat{V}_{AND}, \widehat{V}_{OR})$ con \widehat{G} digrafo conexo simétrico bipartito sin loops tal que el conjunto de puntos fijos de \widehat{N} contiene a S .*

Sea $N = (G, V_{AND}, V_{OR})$ una red AND-OR con G digrafo conexo sin loops. Por Teorema 3 existe una red AND-OR $\widehat{N} = (\widehat{G}, \widehat{V}_{AND}, \widehat{V}_{OR})$ con \widehat{G} digrafo conexo simétrico bipartito sin loops tal que el conjuntos de nodos \widehat{V}_{AND} y \widehat{V}_{OR} son ISs, y todos los puntos fijos de N son puntos fijos de \widehat{N} (ejemplo). Así, es suficiente estudiar el máximo número de puntos fijos de redes AND-OR teniendo estas propiedades para conocer el máximo número de puntos fijos de cualquier red AND-OR.

En adelante, por simplicificación de la notación, asumiremos que cualquier red AND-OR N tiene las mismas propiedades que \widehat{N} .

Definición 2 *Dado $N = (G, V_{AND}, V_{OR})$ y $x \in \{0, 1\}^n$ un punto fijo de N , se definen los conjuntos $V_{OR}^1(x)$ y $V_{OR}^0(x)$ por*

$$V_{OR}^1(x) = \{j \in V_{OR} \mid f_j(x) = 1\}, \quad V_{OR}^0(x) = \{j \in V_{OR} \mid f_j(x) = 0\}.$$

Análogamente se definen los conjuntos: $V_{AND}^1(x)$ y $V_{AND}^0(x)$.

Proposition 1 *Sea $N = (G, V_{AND}, V_{OR})$ y $x \in \{0, 1\}^n$ un fijo punto de N , entonces el conjunto de nodos $V_{OR}^0(x) \cup V_{AND}^1(x)$ es un MIS.*

Demostración. Suponga que existen nodos $i, j \in V_{OR}^0(x) \cup V_{AND}^1(x)$ tales que $\{i, j\} \in E$. Si $i \in V_{OR}^0(x)$, entonces $i \in V_{OR}$ y $f_i(x) = 0$, y ya que el conjunto de nodos V_{OR} es un

IS, $j \in V_{AND}^1(x)$, de aquí, $f_j(x) = 1$, lo cual es una contradicción. El caso $i \in V_{AND}^1(x)$ es análogo. Por lo tanto, $V_{OR}^0(x) \cup V_{AND}^1(x)$ es un IS.

Ahora suponga que existe un nodo $k \in V_{OR}^1(x) \cup V_{AND}^0(x)$ tal que $\{k\} \cup V_{OR}^0(x) \cup V_{AND}^1(x)$ es un IS. Si $k \in V_{OR}^1(x)$, entonces por definición $k \in V_{OR}$ y $f_k(x) = 1$, de aquí existe $j \in N_-(k)$ tal que $f_j(x) = x_j = 1$. Ya que V_{OR} es un IS, $j \in V_{AND}$, así $j \in V_{AND}^1(x)$ lo que es contradictorio con el hecho que $\{k\} \cup V_{AND}^1(x)$ es un IS. El case $k \in V_{AND}^0(x)$ es análogo. Por lo tanto, $V_{OR}^0(x) \cup V_{AND}^1(x)$ es un MIS. ■

Proposition 2 Dado $N = (G, V_{AND}, V_{OR})$, sean $V_0 \subseteq V_{OR}$ y $V_1 \subseteq V_{AND}$ dos conjuntos de nodos tal que $V_1 \cup V_0 \neq \emptyset$. Suponga $V_0 \cup V_1$ un MIS, y sea $x \in \{0, 1\}^n$ el vector definido por

$$x_i = 0, \forall i \in V_0 \cup (V_{AND} \setminus V_1) \quad y \quad x_i = 1, \forall i \in V_1 \cup (V_{OR} \setminus V_0).$$

Entonces x es un punto fijo de N .

Demostración. Suponga que $i \in V_0 \cup (V_{AND} \setminus V_1)$, así por definición $x_i = 0$. Si $i \in V_0 \subseteq V_{OR}$, entonces ya que V_{OR} es un IS, para todo $j \in \Gamma_G^-(i)$ $j \in V_{AND}$, y ya que $V_0 \cup V_1$ es un MIS, para todo $j \in \Gamma_G^-(i)$ $j \in V_{AND} \setminus V_1$. Así, para todo $j \in \Gamma_G^-(i)$ $x_j = 0$ y por lo tanto $f_i(x) = OR(x_j, j \in \Gamma_G^-(i)) = 0 = x_i$. Si $i \in (V_{AND} \setminus V_1)$, entonces existe $j \in \Gamma_G^-(i) \cap V_0$, ya que de lo contrario el conjunto $\{j\} \cup V_0 \cup V_1$ sería un IS, lo cual es una contradicción con $V_0 \cup V_1$ un MIS. De aquí, $x_j = 0$, y por lo tanto $f_i(x) = f(\dots x_j \dots) = 0 = x_i$. Una conclusión análoga puede ser construída para $i \in V_1 \cup (V_{OR} \setminus V_0)$. Por lo tanto, x es un punto fijo de N . ■

De esta manera, dada una red AND-OR $N = (G, V_{AND}, V_{OR})$ se construye la siguiente función

$$H_N : \{x \in \{0, 1\}^n \mid x \text{ es punto fijo de } N\} \longrightarrow \{F \subseteq G \mid F \text{ es un MIS de } G\}$$

definido por $H_N(x) = V_{OR}^0(x) \cup V_{AND}^1(x)$.

Observe que, por Propositiones 1 y 2, H_N es una biyección. Además, $H_N(\vec{1}) = V_{AND}$ y $H_N(\vec{0}) = V_{OR}$.

Por lo tanto, el máximo número de puntos fijos de las redes AND-OR es igual al máximo número de MISs en digrafos conexos simétricos o equivalentemente grafos conexos y además bipartitos. Este resultado permite probar el siguiente teorema.

Teorema 4 Dada $N = (G, V_{AND}, V_{OR})$ una red AND-OR con G digrafo conexo sin loops. Si $|V(G)| = n$, entonces el número de puntos fijos de N es a lo más $2^{(n-1)/2}$ para n impar, y $2^{(n-2)/2} + 1$ para n par.

Demostración. Por Teorema 3 podemos suponer que G es un grafo conexo bipartito y tal que el conjuntos de nodos V_{AND} y V_{OR} son ISs. Por Propositiones 1 y 2 el número de puntos fijos de N es igual al número de MISs de G . Por otro lado, Jiuquiang Liu in [8] probó que el mayor número de MISs que cualquier grafo no-dirigido conexo bipartito con n nodos puede tener es $2^{(n-1)/2}$ para n impar, y $2^{(n-2)/2} + 1$ para n par; y este

número es alcanzado en para una familia dada de grafos. ■

5. Redes Neuronales Discretas (DNN)

Las redes neuronales fueron introducidas por McCulloch y Pitts (1943) para modelar el sistema nervioso. En estas redes el conjunto de estados es $Q = \{0, 1\}$ o equivalentemente $Q = \{-1, 1\}$ que representan el estado de inhibición y activación de cada nodo respectivamente.

Las funciones de activación local, llamadas funciones umbrales, reciben como argumento la suma ponderada de las entradas. Si esta suma es mayor que un cierto valor crítico llamado umbral y que depende de cada nodo, entonces el resultado es la activación del nodo e inactivación en caso contrario. Esto queda representado por la siguiente función:

$$f_i = \text{sign}\left(\sum_{j=1}^n w_{ij}x_j - b_i\right), \quad x \in \{-1, 1\}^n$$

donde $\text{sign}(u) = 1$ si $u \geq 0$ y $\text{sign}(u) = -1$ de lo contrario. En el caso $Q = \{0, 1\}$, la función $\text{sign}(u)$ es reemplazada por la función de Heaviside $\mathbb{1}(u)$, donde $\mathbb{1}(u) = 1$ si $u \geq 0$ y 0 de lo contrario. Así, una red neural discreta \mathcal{N} , actuando sobre el conjunto de estados $\{-1, 1\}$ ó $\{0, 1\}$, es una tupla $\mathcal{N} = (G, W, b, h)$, donde W es una matriz (n, n) -real, G es el grafo de conexión asociado a W tal que $(v_j, v_i) \in E \iff w_{ij} \neq 0$; b es un vector real umbral y $h = \mathbb{1}$ ó $h = \text{sign}$ es el tipo de las funciones de activaciones locales, tal que para cada nodo v_i hay una función dada por:

$$f_i(x) = \mathbb{1}\left(\sum_{j=1}^n w_{ij}x_j - b_i\right), \quad x \in \{0, 1\}^n$$

ó

$$f_i(x) = \text{sign}\left(\sum_{j=1}^n w_{ij}x_j - b_i\right), \quad x \in \{-1, 1\}^n$$

respectivamente, donde $x = (x_1, x_2, \dots, x_n)$ denota el vector de estados de la red.

La actualización de los valores de estado de cada nodo depende del tipo de iteración asociado al modelo. La más común es la iteración paralela consistente en actualizar el valor de todos los nodos al mismo tiempo, es decir,

$$\forall i \in \{1, \dots, n\}, \quad x_i(t+1) = f_i(x_j(t), j \in N_-(i)).$$

Las demostraciones de los teoremas y resultados complementarios de esta sección pueden ser encontrados en [10,6,2]

5.1. DNNs simétricas y Funcionales de Lyapunov.

Sea $\mathcal{N} = (G, W, b, \mathbb{1})$ una DNN y sea $x(t) \in \{0, 1\}^n$ un vector de configuración de estados. Se define el funcional de Lyapunov:

$$E(x(t+1)) = - \sum_{i \in V} \sum_{j \in V} w_{ij} x_i(t+1) x_j(t) + \sum_{i \in V} b_i (x_i(t+1) + x_i(t)).$$

Proposición 4 *Sea $\mathcal{N} = (G, W, b, \mathbb{1})$ una DNN con W una matriz simétrica. Entonces $E(x(t+1))$ definido previamente es un funcional de Lyapunov estrictamente decreciente.*

Demostración. Como W es simétrica se tiene que:

$$\Delta E = E(x(t+1)) - E(x(t)) = - \sum_{i \in V} (x_i(t+1) - x_i(t-1)) \left(\sum_{j \in V} w_{ij} x_j(t) - b_i \right).$$

Por otro lado, de la definición de iteración paralela se tiene que:

$$-(x_i(t+1) - x_i(t-1)) \left(\sum_{j \in V} w_{ij} x_j(t) - b_i \right) \leq 0.$$

Así se tiene que $\Delta E \leq 0$ y $\Delta E < 0$ si y sólo si $x_i(t+1) \neq x_i(t-1)$. ■

Corolario 2 *Sea $\mathcal{N} = (G, W, b, \mathbb{1})$ una DNN con W una matriz simétrica. Entonces, los atractores de \mathcal{N} iterada en forma paralela son sólo puntos fijos o ciclos límites de largo dos.*

De igual forma, funcionales de Lyapunov pueden ser construídos para determinar una cota superior del largo de los transientes de DNNs iteradas en forma paralela.

Proposición 5 *Sea $\mathcal{N} = (G, W, b, \mathbb{1})$ una DNN con W una matriz simétrica. Entonces el largo máximo de los transientes de la red iterada en forma paralela $\tau_p(\mathcal{N})$ satiface:*

$$\tau_p(\mathcal{N}) \leq \frac{1}{2} (\|A\| + 3\|2b - A\mathbb{1}\| - n).$$

Sin embargo, Goles y Cosnard caracterizaron la familia de DNNs que admiten un funcional de Lyapunov y de paso probaron que sólo una fracción pequeña de DNNs admite un tal funcional. Esta familia corresponde a aquella donde la matriz W es cuasi-simétrica, es decir, $\forall i, j \in V, \exists \lambda_i, \lambda_j > 0$ tal que $\lambda_i w_{ij} = \lambda_j w_{ji}$.

5.2. Ciclos Positivos y Negativos en Redes Neuronales Discretas

Sea $\mathcal{N} = (G, W, b, \text{sign})$ una DNN. Definimos el signo de un arco (v_i, v_j) , denotado por $\text{sign}((v_i, v_j))$, como el signo de w_{ji} . El signo de un camino C (denotado por $\text{sign}(C)$),

con respecto a W , es positivo si el número de arcos negativos de C es par y negativo de lo contrario. Llamaremos un ciclo C negativo (respectivamente positivo) si $\text{sign}(C)$ es negativo (respectivamente positivo).

En lo que sigue, asumiremos que el grafo G es conexo, sino se puede aplicar los resultados a cada componente conexa de G .

Además, supondremos sin pérdida de generalidad que $d_-(v_i) > 0$ for all $v_i \in V$. Pues de lo contrario, si existe un nodo $v_i \in V$ tal que $d_-(v_i) = 0$, entonces la función f_i es constante, y este caso no es interesante. Evidentemente sigue que existe al menos un ciclo C en G . Finalmente, supondremos que el grafo G y la matriz W tienen una estructura quasi-minimal, esto es $\forall (v_i, v_j) \in E, i \neq j, d_-(v_i) > 1, \exists x \in \{-1, 1\}^n$, tal que

$$\text{sign}\left(\sum_k w_{jk}x_k - b_j\right) \neq \text{sign}\left(\sum_{k \neq i} w_{jk}x_k - b_j\right) \quad (1)$$

Es decir, en una red con estructura quasi-minimal si es removido cualquier arco de G , entonces la dinámica de la red cambia.

De aquí, una condición para tener estructura quasi-minimal es tener funciones no constantes, lo cual implica que:

$$-\sum_j |w_{ij}| < b_i \leq \sum_j |w_{ij}| \quad \forall i = 1, \dots, n \quad (2)$$

La siguiente es una propiedad de caracterización de los ciclos positivos y negativos en una red \mathcal{N} .

Proposition 3 *Dada $\mathcal{N} = (G, W, b, \text{sign})$ una DNN, un ciclo C en \mathcal{N} es positivo si y sólo si existe un vector $x \in \{-1, 1\}^{l(C)}$ tal que x y $-x = (-x_1, \dots, x_n)$ satisfacen la ecuación:*

$$\forall (v_i, v_j) \in C, \quad \text{sign}(w_{ji}) = x_i \cdot x_j \quad (3)$$

o equivalentemente

$$\forall (v_i, v_j) \in C, \quad x_j = \text{sign}(w_{ji}) \cdot x_i$$

Demostración. Sea $C = v_0, v_1, \dots, v_{k-1}, v_0$ un ciclo positivo en \mathcal{N} . definamos el vector $x \in \{-1, 1\}^k$ como sigue:

$$\bullet \quad x_0 = 1 \text{ y } x_j = \begin{cases} \text{sign}(w_{jj-1}) \cdot x_{j-1} & \text{si } (v_{j-1}, v_j) \in C \\ \text{sign}(w_{j-1j}) \cdot x_{j-1} & \text{si } (v_j, v_{j-1}) \in C \end{cases}, \forall j = 1 \dots, k$$

Así x satisface la ecuación 3. De aquí, $-x$ también satisface la ecuación 3. Finalmente, es fácil chequear que x y $-x$ son los únicos vectores satisfaciendo la ecuación 3.

Ahora, sea C ciclo negativo, y supongamos que la ecuación 3 es verdadera, entonces

$$\prod_{(v_i, v_j) \in C} \text{sign}(w_{ij}) = \left(\prod_{j \in C} x_j \right) \cdot \left(\prod_{i \in C} x_i \right) = \left(\prod_{j \in C} x_j \right)^2$$

pero $\text{sign}(C) = \prod_{(v_i, v_j) \in C} \text{sign}(w_{ij}) < 0$, lo cual es una contradicción. \blacksquare

Teorema 5 Dado $\mathcal{N} = (G, W, b, \text{sign})$ una DNN, si todos los ciclos del grafo de conexión G son positivos, entonces existe un vector $x = (x_1, \dots, x_n) \in \{-1, 1\}^n$ tal que x y $-x = (-x_1, \dots, -x_n)$ son puntos fijos de \mathcal{N} .

La condición del teorema previo no es una condición necesaria como se muestra en el siguiente ejemplo:

Ejemplo 1 Sea $\mathcal{N} = (G, W, b, \mathbb{1})$ una DNN donde G es el grafo de la Figura 9a), los valores de la matriz W son dados por la Figura y $b = (0, 0, 0, 0, 0)$. \mathcal{N} contiene un ciclo negativo $\{(3, 4), (4, 5), (5, 3)\}$. Sin embargo, \mathcal{N} tiene dos puntos fijos: $x = (1, 1, 1, 1, 1)$ y $-x = (-1, -1, -1, -1, -1)$.

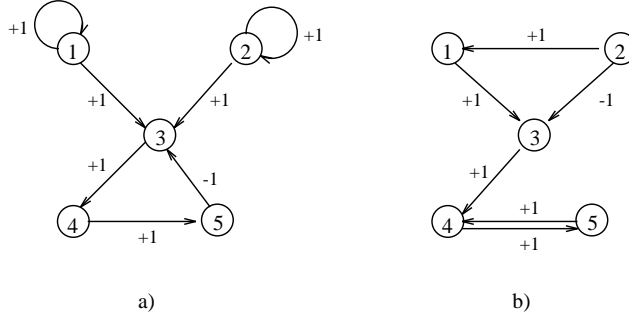


Figura 9. Ejemplo de grafos teniendo ciclos positivos y negativos.

El siguiente teorema nos da una condición necesaria para tener puntos fijos.

Teorema 6 Dada $\mathcal{N} = (G, W, b, \text{sign})$ una DNN, si todos los ciclos de G son negativos, entonces \mathcal{N} no tiene puntos fijos.

Demostración. Sea $x \in \{-1, 1\}^n$ un punto fijo de \mathcal{N} y C un ciclo. Como C es negativo, por Propiedad 3 existen $(v_i, v_j) \in C$ tal que $\text{sign}(w_{ji})x_i \neq x_j$. denotemos por E^- el conjunto de arcos, definido por $E^- = \{(v_i, v_j) \in E \mid \text{sign}(w_{ji})x_i \neq x_j\}$, el cual es no vacío. Además, denotemos por \mathcal{N}' la DNN definida por $\mathcal{N}' = (G', W', b, \text{sign})$, donde $G' = (V, E \setminus E^-)$.

Si para todo $v_j \in V$, $d_-(v_j) > 0$, entonces existe un ciclo en G' el cual por hipótesis es negativo, lo que es imposible por construcción de G' , por lo tanto existe un nodo $v_{j'}$ tal que $N_-(v_{j'}) = \emptyset$; esto significa que $\text{sign}(w_{j'k})x_k \neq x_{j'}$, $\forall k \in N_-(v_{j'})$ y por lo tanto si $x_{j'} = 1$, entonces

$$\sum_{k \in N_-(v_{j'})} w_{j'k} x_k = - \sum_{k \in N_-(v_{j'})} |w_{j'k}|$$

lo cual implica que

$$\sum_{k \in N_-(v_{j'})} w_{j'k} x_k - b_{j'} < 0$$

así $x_{j'}$ no es una componente fija de x lo cual es una contradicción. El mismo argumento puede ser aplicado en el caso de $x_{j'} = -1$. ■

Es fácil chequear que la condición del teorema previo no es una condición suficiente (ejercicio)

5.3. Máximo número de puntos fijos en DNNs.

Los resultados de la sección anterior muestran una relación entre los ciclos positivos del grafo y la existencia de puntos fijos en la red. En esta sección mostraremos que existe una función monótona creciente entre ambos conjuntos.

Previamente algunas definiciones.

Definición 3 Llamaremos conjunto de nodos recubridor de ciclos positivos (PFVS) de un grafo $G = (V, E)$ a un conjunto $T \subseteq V$, tal que

$$T \cap C \neq \emptyset, \quad \forall C \in \mathcal{C}^+(G),$$

donde $\mathcal{C}^+(G)$ es el conjunto de ciclos positivos de G .

el mínimo número de vértices de un PFVS de G será denotado por:

$$\tau(G) = \min |T|.$$

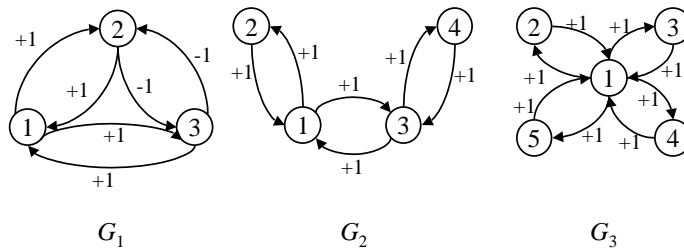


Figura 10. Aquí $\tau(G_1) = \tau(G_2) = 2$ y $\tau(G_3) = 1$.

Teorema 7 Dada \mathcal{N} DNN, el número de puntos fijos de \mathcal{N} es menor o igual a $2^{\tau(G)}$, y esta cota es alcanzada.

Así, la estructura de un grafo G determina el máximo número de puntos fijos para todas las DNNs \mathcal{N} teniendo su grafo de conexión G independientemente del valor de sus parámetros. Un ejemplo de esto es \mathcal{N} con un grafo de conexión igual a una estrella simétrica (ver Figura 11).

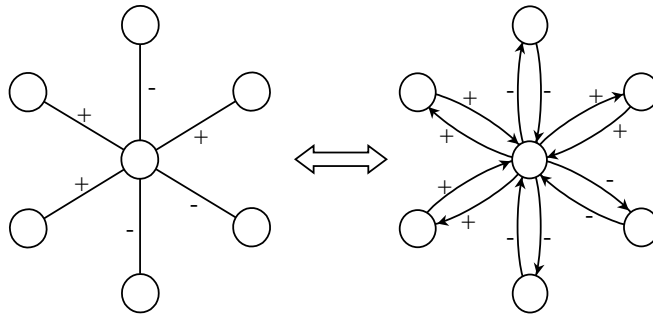


Figura 11. Ejemplo de digrafo estrella. Para este digrafo sólo es posible construir una red \mathcal{N} con a lo más dos puntos fijos.

Referencias

- [1] Goles E. Aracena J., Demongeot J. Fixed points and maximal independent sets on and-or networks. *Discrete Applied Mathematics*, 138:277–288, 2004.
- [2] Goles E. Aracena J., Demongeot J. Positive and negative circuits in discrete neural networks. *IEEE Transactions on Neural Networks*, 1:77–83, 2004.
- [3] J. H. Conway. Unpredictable iterations. In *Number Theory Conference*, pages 49–52, Boulder, Colorado, 1972. University of Colorado.
- [4] A. K. Dewdney. Computer recreations: Two-dimensional turing machines and tur-mites make tracks on a plane. *Scientific American*, pages 124–127, September 1989.
- [5] A. K. Dewdney. Computer recreations:.. *Scientific American*, pages 118–121, March 1990.
- [6] Goles E. and Martínez S. *Neural and Automata Networks*. Kluwer Academic Press, 1991.
- [7] A. Gajardo, A. Moreira, and E. Goles. Complexity of langton’s ant. *Discrete Applied Mathematics*, 117:41–50, 2002.
- [8] Liu J. Maximal independent sets in bipartite graphs. *J. Graph Theory*, 17:495–507, 1993.
- [9] C. G. Langton. Studing artifitial life with cellular automata. *Physica D*, 22:120–149, 1986.
- [10] Goles E. Cosnard M. Discrete state neural networks and energies. *Neural Networks*, 10:327–333, 1997.
- [11] Kauffman S. Metabolic stability and epigenesis in randomly constructed genetics nets. *J. Theoretical Biology*, 22:437–467, 1969.