

Universidad del Bío-Bío
Facultad de Ciencias Empresariales
Magíster en Ciencias de la Computación



UNIVERSIDAD DEL BÍO-BÍO

Toolbox de binSMART
Estudio de su Dinámica

Estudiante: Reynaldo Andrés Rivas Oliva
Guía: Rodrigo Torres Avilés
Examen de grado: 2026

Universidad del Bío-Bío

Índice general

Resumen	iv
Abstract	1
1 Introducción	2
2 Estado del arte	5
2.1 Definiciones Preliminares	5
2.1.1 Máquinas de Turing	5
2.1.2 Sistemas dinámicos	7
2.1.3 Velocidad en MT	8
2.1.4 Las máquinas SMART y binSMART	9
2.1.5 Comportamiento de ida y vuelta (zigzag)	11
2.1.6 Weak Mixing	12
2.1.7 Universalidad física	12
2.2 Discusión	13
La construcción de la toolbox de SMART como metodología a replicar	13
El paper original de binSMART y su Lemma 1	14
El paper de Kůrka y el origen de la conjetura	15
El antecedente del speed positiveness problem	16
3 Definición del problema	17
3.1 Alcance	17
4 Objetivos e hipótesis	19
4.1 Objetivos de la Investigación	19
4.1.1 Objetivo General	19
4.1.2 Objetivos Específicos	19

4.2	Hipótesis	19
5	Metodología	21
6	Resultados	23
6.1	Construcción de la toolbox de binSMART	23
6.1.1	Movimientos de escala n : D'_1 y D_3	23
6.1.2	Movimientos compuestos: D_1 y G_1	26
6.1.3	Movimientos de transición: D'_3 y G'_3	27
6.2	Análisis de configuraciones iniciales: velocidad de binSMART	28
6.2.1	Estados que alcanzan la posición -1	29
6.2.1.1	Estado inicial $1D'$	29
6.2.1.2	Estado inicial $3G$	29
6.2.1.3	Estado inicial $3D'$	29
6.2.1.4	Estado inicial $1G$	30
6.2.1.5	Estado inicial $1D$	31
6.2.2	Estados descartados por ser más lentos que $Q(k)$	32
6.2.2.1	Estado inicial $3D$	32
6.2.3	Estado inicial $3G'$: configuración óptima	33
6.2.3.1	Estado inicial $1G'$	34
6.2.4	Configuraciones que descubren celdas en ambas direcciones	35
7	Discusión	39
7.1	Sobre la construcción de la toolbox	39
7.2	Sobre la velocidad logarítmica de binSMART	40
7.3	Sobre el camino abierto hacia weak mixing	41
8	Conclusiones	42
8.1	Síntesis de resultados	42
8.2	Alcance y limitaciones	43
8.3	Recomendaciones para trabajo futuro	43

Índice de figuras

2.1	Diagrama de estados y transiciones de la máquina SMART, con cuatro estados (b, d, p, q) sobre el alfabeto $\{0, 1, 2\}$	10
2.2	Diagrama de estados y transiciones de la máquina binSMART, con ocho estados sobre el alfabeto binario $\{0, 1\}$	11

Resumen

Esta tesis desarrolla una toolbox analítica y cuantitativa para la máquina de Turing binaria binSMART, una reconstrucción de dos símbolos de la máquina SMART. La caracterización previa de sus movimientos, dada en el Lemma 1 de Concha-Vega y Torres-Avilés, es cualitativa y no cuantifica el tiempo exacto de cada patrón de búsqueda. Siguiendo la metodología aplicada por Torres-Avilés sobre SMART, esta tesis extiende ese marco mediante fórmulas cerradas de tiempo exacto, demostradas por inducción sobre las recurrencias asociadas.

A partir de esta toolbox se determina la velocidad de la máquina binSMART, mediante el tiempo mínimo

$$Q(k) = 3 + T(k-4) = \frac{1}{6} \left(-(-1)^{k-3} + 2^{k+1} + 3 \right)$$

necesario para que una configuración inicial descubra k celdas nuevas de la cinta. También se identifican las configuraciones óptimas y se demuestra que ninguna trayectoria *unbounded* puede igualar dicho tiempo. En consecuencia, el número máximo de celdas nuevas descubiertas en tiempo t satisface $m(t) = \Theta(\log_2 t)$, por lo que binSMART constituye un ejemplo explícito de máquina mínima aperiódica con crecimiento logarítmico dentro del régimen distorsionado. Estos resultados abren además el camino para estudiar weak mixing y mixing topológico en binSMART.

Abstract

This thesis develops an analytical and quantitative toolbox for the binary Turing machine binSMART, a two-symbol reconstruction of the SMART machine. The prior characterization of its movements, given in Lemma 1 by Concha-Vega and Torres-Avilés, is qualitative and does not quantify the exact time of each bounded-search pattern. Following the methodology applied by Torres-Avilés to SMART, this thesis extends that framework through closed-form exact-time formulas, proved by induction over the associated recurrences.

Using this toolbox, the speed of the binSMART machine is determined, via the minimum time

$$Q(k) = 3 + T(k-4) = \frac{1}{6} \left(-(-1)^{k-3} + 2^{k+1} + 3 \right)$$

required for an initial configuration to discover k new tape cells. The optimal configuration families are identified, and it is shown that no *unbounded* trajectory can match this time. Consequently, the maximum number of new cells discovered in time t satisfies $m(t) = \Theta(\log_2 t)$, so binSMART provides an explicit example of a minimal aperiodic machine with logarithmic growth in the distorted regime. These results also open the way to future work on weak mixing and topological mixing for binSMART.

Capítulo 1

Introducción

En 1936 se formalizó la noción de máquina de cómputo como una idealización del proceso mediante el cual un ser humano realiza cálculos siguiendo un conjunto finito de reglas mecánicas, introduciendo el modelo de cabezal, cinta y m-configuraciones que hoy se conoce como máquina de Turing [16]. En dicho trabajo se demostró que no existe un proceso general capaz de determinar si una máquina imprime infinitos símbolos, y a partir de esta indecidibilidad, mediante la construcción de una máquina universal, que el *Entscheidungsproblem* de Hilbert no admite solución general [16]. Durante más de medio siglo, la investigación sobre máquinas de Turing se concentró casi exclusivamente en esta pregunta clásica, relativa a qué puede computar una máquina y con qué recursos de tiempo o espacio.

No obstante, toda máquina de Turing define asimismo un proceso determinista que evoluciona sobre un espacio de configuraciones, proceso que puede estudiarse en sí mismo, con independencia del resultado que produzca. En 1997 se formalizó esta perspectiva mediante dos modelos topológicos que resuelven el problema de la no compacidad del espacio de configuraciones, denominados modelo de cinta móvil (TMT) y modelo de cabezal móvil (TMH) [11]. Bajo el modelo TMT se conjeturó que toda máquina de Turing no trivial requiere, para evitar la periodicidad temporal, desplazarse sobre distancias arbitrariamente largas, de modo que la existencia de al menos un punto periódico constituiría una consecuencia necesaria de dicho modelo [11]. Este trabajo dio origen a una línea de investigación que combina la computabilidad con la dinámica topológica y simbólica, en la cual propiedades como la periodicidad, la transitividad, la entropía y la velocidad de desplazamiento del cabezal se constituyen en objetos de estudio legítimos.

Esta conjetura fue refutada mediante la construcción de la primera máquina de Turing aperiódica, empleando una técnica inspirada en las llamadas recursivas introducidas en 1966 para resolver el problema de la inmortalidad, consistente en decidir si existe una configuración desde la cual la máquina nunca alcanza un estado dado ni se detiene, problema que también resultó ser indecidible [8, 2]. Dicha máquina no resultó ser reversible, lo cual dejó abierta la pregunta acerca de la existencia de una máquina simultáneamente aperiódica, completa y reversible. Posteriormente se demostró la indecidibilidad de la aperiodicidad para máquinas reversibles no completas, planteándose explícitamente esta pregunta como

conjetura abierta [10].

La respuesta afirmativa a esta conjetura fue proporcionada por la máquina SMART, de cuatro estados y tres símbolos, la cual resulta ser completa, reversible, aperiódica, simétrica en el tiempo y topológicamente minimal, constituyendo el ejemplo más pequeño conocido con esta combinación de propiedades [3]. Su aperiodicidad se sustenta en un Lemma 1 que certifica, mediante inducción, la existencia de cuatro patrones de búsqueda acotada, sin cuantificar el número de pasos que requiere cada uno de ellos [3]. A partir de esta construcción se demostró, mediante una técnica de *embedding*, que determinar si una máquina reversible y completa arbitraria posee un punto periódico es indecidible [3]. Esta misma técnica permitió posteriormente probar la indecidibilidad de la positividad de la entropía topológica [6], en contraste con su computabilidad aproximada [9], y más adelante la indecidibilidad de la positividad de la velocidad máxima [15].

Dado que la máquina SMART se encuentra definida sobre un alfabeto ternario, surgió la pregunta de si existía una reconstrucción binaria capaz de preservar su dinámica. Esta pregunta fue respondida en 2021 mediante la construcción de binSMART, la cual duplica el número de estados internos, de cuatro a ocho, con el propósito de codificar en binario la información que SMART almacenaba en su tercer símbolo [4]. binSMART conserva las propiedades de completitud, reversibilidad, aperiodicidad y minimalidad topológica de su predecesora, y su t-shift resulta ser sustitutivo, aunque pierde la simetría temporal característica de SMART [4]. Al igual que en SMART, su aperiodicidad descansa sobre un Lemma 1 que certifica la validez de ocho movimientos elementales de manera puramente existencial, sin registrar el número de pasos que requiere cada transición [4].

La máquina binSMART constituye el objeto de estudio central de esta tesis. Para SMART se demostró, en 2022, que su sistema TMT asociado es topológicamente débilmente mezclante (*weak mixing*); sin embargo, no se conoce si binSMART conserva dicha propiedad [14, 4]. Esta pregunta abierta constituye el punto de partida del presente trabajo. La demostración de weak mixing en SMART se apoyó en lemas auxiliares que cuantifican con exactitud el número de pasos requerido para sincronizar configuraciones dentro de una ventana de tiempo común, y recurrió nuevamente a la técnica de embedding para demostrar que la mezcla débil, la mezcla topológica y la transitividad total resultan indecidibles en general para máquinas reversibles y completas [14], en paralelo a un resultado análogo obtenido para autómatas celulares reversibles [12]. Para binSMART, en cambio, no existía hasta la fecha una toolbox de movimientos con dicha cuantificación temporal explícita.

La contribución de esta tesis consiste en la construcción de dicha toolbox cuantitativa, extendiendo el Lemma 1 de [4] mediante fórmulas cerradas, demostradas por inducción, para el costo temporal exacto de los ocho movimientos elementales de binSMART, siguiendo la metodología aplicada posteriormente sobre SMART para demostrar su weak mixing [14]. A partir de esta toolbox se aborda el problema de determinar el tiempo mínimo global requerido para que una configuración inicial descubra

una cantidad arbitraria de celdas nuevas de la cinta, con el propósito de identificar las familias de configuraciones óptimas asociadas a dicho mínimo. Este análisis se extiende, asimismo, a las trayectorias zigzag propias de binSMART, en las cuales el cabezal alterna de dirección cada vez que escribe o encuentra un símbolo 1, de modo que la anchura de cada tramo recorrido no es fija ni necesariamente creciente, a diferencia de la noción de zigzag basada en una estructura de pila empleada en [5] para caracterizar la complejidad del lenguaje del t-shift asociado a una máquina. El propósito de esta extensión es establecer si tales trayectorias son capaces de igualar el tiempo de la configuración óptima unidireccional. Estos resultados, desarrollados en el Capítulo de *Resultados*, no resuelven por sí mismos la pregunta acerca de si binSMART es weak mixing, pero constituyen el fundamento cuantitativo sobre el cual dicha pregunta podría abordarse en trabajo futuro.

El presente informe se organiza de la siguiente manera. El Capítulo de *Estado del arte* presenta las definiciones formales y los antecedentes metodológicos de esta tesis. El Capítulo de *Objetivos e hipótesis* formula el objetivo general, los objetivos específicos y las hipótesis de investigación. El Capítulo de *Metodología* describe la planificación temporal del trabajo y el cumplimiento de cada objetivo específico. El Capítulo de *Resultados* contiene la construcción de la toolbox de movimientos de binSMART y el análisis de la configuración óptima, incluyendo su extensión a trayectorias zigzag. Finalmente, los capítulos de *Discusión* y *Conclusiones* sitúan estos resultados en el contexto de la literatura revisada y proponen líneas de trabajo futuro, entre ellas el estudio directo de la propiedad de weak mixing en binSMART.

Capítulo 2

Estado del arte

2.1 Definiciones Preliminares

2.1.1 Máquinas de Turing

Definición 2.1 (Máquina de Turing clásica). Una máquina de Turing, en su formulación clásica [16], es una quintupla

$$M = (Q, A, q_0, q_f, \delta),$$

donde Q es un conjunto finito de estados (m-configuraciones), A es un alfabeto finito de símbolos que incluye un símbolo distinguido $S_0 \in A$ llamado *blanco*, $q_0 \in Q$ es el estado inicial, $q_f \in Q$ es el estado final o de detención, y δ es la función de transición, definida como una función parcial sobre $Q \setminus \{q_f\}$. El requisito de un símbolo blanco S_0 permite representar entradas finitas mediante cintas que son blancas salvo en un número finito de casillas; el estado inicial q_0 fija el punto de partida del cómputo, y el estado final q_f determina su término, considerándose que la máquina se detiene si y solo si alcanza dicho estado.

Definición 2.2 (Configuración completa). Dada una máquina de Turing $M = (Q, A, q_0, q_f, \delta)$, una *configuración completa* en una etapa dada es la terna

$$c = (i, w, q),$$

donde $i \in \mathbb{Z}$ es la posición de la casilla explorada, $w \in A^{\mathbb{Z}}$ es el contenido completo de la cinta y $q \in Q$ es el estado vigente. Un *movimiento* de M es la transición de una configuración completa c a la siguiente, determinada por la aplicación de δ sobre el par (q, w_i) .

La formalización clásica exige un alfabeto con blanco, un estado inicial y un estado final, condiciones orientadas a modelar el cómputo de una función sobre una entrada finita. Sin embargo, cuando el objetivo es estudiar la evolución de la máquina como un sistema dinámico, estas tres condiciones dejan de ser necesarias: no hay una entrada que delimitar (la cinta puede ser bi-infinita desde el inicio), no hay un cómputo que iniciar en un estado privilegiado (la evolución se estudia a partir de cualquier configuración),

y no hay un resultado que aceptar mediante un estado final (el objeto de interés es el comportamiento a largo plazo, no la detención). Esto motiva la siguiente definición, adoptada en esta tesis y en la literatura de sistemas dinámicos sobre máquinas de Turing [11, 3, 4, 14].

Definición 2.3 (Máquina de Turing, definición dinámica). Una máquina de Turing es una tripleta

$$M = (Q, A, \delta),$$

donde Q es un conjunto finito de estados, A es un alfabeto finito de símbolos, y

$$\delta : Q \times A \rightarrow Q \times A \times \{-1, 0, 1\}$$

es la función de transición. Se dice que M es *determinista* si δ es funcional, y *completa* si δ es total en $Q \times A$. Una configuración de M es una terna $(w, i, q) \in A^{\mathbb{Z}} \times \mathbb{Z} \times Q$, y la aplicación de una instrucción $\delta(q, w_i) = (q', s', d)$ transforma dicha configuración escribiendo s' en la posición i , desplazando la posición explorada a $i + d$ y actualizando el estado a q' .

A diferencia de la definición clásica, esta formulación no distingue estados iniciales ni finales, ni exige un símbolo blanco: toda configuración en $A^{\mathbb{Z}} \times \mathbb{Z} \times Q$ es admisible como punto de partida, y la máquina evoluciona indefinidamente salvo que alcance un par (q, s) fuera del dominio de δ , en cuyo caso se dice que M se detiene. Es esta definición dinámica, y no la clásica, la que se emplea en el resto de este informe.

Modelo de quintuplas y modelo de cuádruplas.

La función de transición δ de una máquina de Turing puede describirse mediante dos modelos de instrucciones distintos, computacionalmente equivalentes entre sí. En el *modelo de quintuplas*, cada instrucción tiene la forma (q, s, q', s', d) , e indica que, estando la máquina en el estado q y leyendo el símbolo s , escribe el símbolo s' y se desplaza en la dirección d en una misma unidad de tiempo, transitando al estado q' ; este es el modelo empleado en la definición dinámica anterior. En el *modelo de cuádruplas*, en cambio, cada instrucción restringe la acción de la máquina a un único paso elemental por unidad de tiempo, siendo de la forma (q, s, s', q') si escribe (sin desplazarse) o de la forma (q, s, d, q') si se desplaza (sin escribir), pero nunca ambas cosas a la vez; en este modelo, cada estado $q \in Q$ se clasifica de manera exclusiva como un *estado de escritura* o como un *estado de movimiento*.

Es posible transformar una máquina del modelo de quintuplas a una del modelo de cuádruplas, y viceversa. De quintuplas a cuádruplas, basta introducir un estado intermedio por cada instrucción, de modo que la escritura y el desplazamiento se ejecuten en dos pasos consecutivos en lugar de uno solo, lo que duplica el tiempo de ejecución de la máquina original. De cuádruplas a quintuplas, basta fusionar

cada par de instrucciones consecutivas de escritura y movimiento en una única instrucción, reduciendo el tiempo de ejecución a la mitad. Ambas transformaciones preservan el comportamiento de la máquina bajo la formalización clásica, pero afectan su dinámica en el sentido de los sistemas dinámicos: al modificar el tiempo relativo entre configuraciones, las nociones cuantitativas de velocidad y de costo temporal exacto de cada movimiento —el objeto de estudio central de esta tesis— dependen del modelo de instrucciones elegido [15]. Por esta razón, el modelo de cuádruplas es el preferido en el estudio de la reversibilidad, ya que su inversa se define de manera directa como otra máquina de cuádruplas, intercambiando el orden de lectura y escritura, mientras que la inversa de una máquina de quintuplas exigiría desplazarse antes de leer el símbolo correspondiente, lo cual no corresponde a una máquina de Turing bien definida bajo dicho modelo [15].

2.1.2 Sistemas dinámicos

Un sistema dinámico topológico es un par (X, f) , donde X es un espacio métrico compacto y $f : X \rightarrow X$ es una función continua [11, 1]. Sobre esta estructura se definen nociones fundamentales como los puntos periódicos ($f^t(x) = x$ para algún $t > 0$), los puntos de equicontinuidad, los atractores y la entropía topológica, que cuantifica la complejidad de las trayectorias del sistema [12, 1]. Dos sistemas dinámicos (X, f) y (Y, g) se dicen *conjugados* si existe un homeomorfismo $h : X \rightarrow Y$ tal que $h \circ f = g \circ h$, lo cual permite identificar sistemas estructuralmente equivalentes aunque se definan sobre espacios distintos [12].

Dentro de la clase general de sistemas dinámicos topológicos, los *sistemas simbólicos* constituyen una subclase particularmente importante, y es precisamente la clase en la que se enmarca esta tesis. Un sistema simbólico se construye a partir de un alfabeto finito A (con al menos dos símbolos) y del espacio $A^{\mathbb{Z}}$ de sucesiones bi-infinitas de símbolos, equipado con la métrica $d(x, y) = 2^{-n}$, donde $n = \min\{|i| : x_i \neq y_i\}$ es la posición de discrepancia más cercana al origen entre x y y [11]. Topológicamente, $A^{\mathbb{Z}}$ es un espacio de Cantor, también llamado discontinuo: informalmente, es un espacio compacto (cerrado y acotado, en el sentido de que toda sucesión de puntos tiene una subsucesión convergente dentro del propio espacio), cero-dimensional (se puede separar cualquier par de puntos distintos mediante conjuntos abiertos y cerrados a la vez, algo que no ocurre, por ejemplo, en un intervalo real) y perfecto (todo punto es límite de otros puntos del espacio, es decir, no existen puntos aislados) [11]. Esta estructura es la misma que exhibe, por ejemplo, el conjunto de Cantor clásico en $[0, 1]$, y resulta natural en $A^{\mathbb{Z}}$ porque cada sucesión bi-infinita puede pensarse como una rama de un árbol infinito de decisiones, una por cada símbolo del alfabeto en cada posición. La dinámica sobre este espacio viene dada por el *shift* $\sigma : A^{\mathbb{Z}} \rightarrow A^{\mathbb{Z}}$, definido por $\sigma(x)_i = x_{i+1}$, y un *subshift* es cualquier subconjunto cerrado y σ -invariante de $A^{\mathbb{Z}}$ [11]. La diferencia esencial con un sistema dinámico general, entonces, es que en un sistema simbólico el espacio de fases no es un continuo (como una variedad o un intervalo), sino un espacio discreto de configuraciones sobre un alfabeto finito, y la dinámica no es un flujo continuo sino un desplazamiento combinatorio; esta

discretización es precisamente lo que hace tratables herramientas como el conteo de palabras, la entropía combinatoria y los argumentos de compacidad basados en cilindros [11].

Para que una máquina de Turing pueda estudiarse dentro de este marco simbólico, es necesario modificar su formalización clásica en al menos dos aspectos. Primero, la estructura de estados iniciales y finales, así como la noción de aceptación o rechazo de una entrada, se descarta por completo: basta con una función de transición $\delta : Q \times A \rightarrow Q \times A \times \{-1, 0, 1\}$, y la computación puede comenzar en cualquier estado y proseguir indefinidamente sin necesidad de distinguir un símbolo en blanco ni restringir el contenido de la cinta a una porción finita [11]. Lo que antes era el resultado de un cómputo (aceptar, rechazar, o detenerse) se reinterpreta dinámicamente como un punto fijo de la evolución. Segundo, y más importante, es necesario resolver el problema de la compacidad del espacio de configuraciones: el espacio natural $Q \times \mathbb{Z} \times A^{\mathbb{Z}}$, que registra el estado, la posición del cabezal y el contenido de la cinta, no es compacto, precisamente porque la posición del cabezal recorre todo $\mathbb{Z}$ sin cota [11].

Este problema se resuelve de dos maneras distintas, dando lugar a dos modelos topológicos no isomorfos entre sí [11]. El primero, *Turing Machine with Moving Tape* (TMT), evita el problema manteniendo el cabezal fijo en el origen y desplazando la cinta completa en cada transición, de modo que el espacio de configuraciones se reduce a $X = Q \times A^{\mathbb{Z}}$, que sí es compacto y homeomorfo a un espacio de Cantor en el sentido descrito anteriormente [11]. El segundo, *Turing Machine with Moving Head* (TMH), en cambio conserva el cabezal móvil, pero logra la compacidad insertando el propio estado del cabezal como un símbolo adicional directamente sobre la cinta, en el alfabeto ampliado $Q \sqcup A$, de modo que la posición del cabezal queda codificada en la configuración misma en lugar de como una coordenada aparte; el espacio resultante $X_1 \subset (Q \sqcup A)^{\mathbb{Z}}$ es entonces un subshift (de hecho, un sistema sófico) del espacio simbólico completo [11]. En ambos casos, la modificación clave es la misma: sacrificar la representación física e intuitiva de la máquina (cabezal moviéndose sobre una cinta fija) a cambio de una representación puramente simbólica y compacta, que es la que permite aplicar sin restricciones el aparato de la dinámica topológica. A partir de este marco se define también el *t-shift* S_T , que registra la secuencia de pares (estado, símbolo leído) observados por el cabezal en cada paso, siendo un factor continuo y sobreectivo del sistema TMT [11].

2.1.3 Velocidad en MT

La velocidad de una máquina de Turing mide qué tan rápido el cabezal se aleja hacia el infinito en la cinta, y ha sido formalizada de más de una manera en la literatura. Una definición clásica establece, para una configuración c , el número $s_n(c)$ de celdas distintas leídas durante los primeros n pasos de la ejecución, a partir del cual se define la velocidad de la configuración como el promedio asintótico $s(c) = \lim_{n \rightarrow \infty} s_n(c)/n$; la velocidad máxima de la máquina se define entonces como $S(M) = \max_c s(c)$ [9]. Bajo esta definición se demuestra que la velocidad máxima y la entropía topológica de una máquina

de una sola cinta son computables con precisión arbitraria, y que dicha velocidad máxima es siempre un número racional, alcanzado por una configuración eventualmente periódica [9]. Esta noción captura la deriva promedio del cabezal, pero no distingue si las celdas leídas en cada paso corresponden a territorio ya conocido o a celdas nunca antes visitadas.

La definición adoptada en esta investigación, propuesta en el estudio de la distorsión en máquinas de un cabezal[7], se aparta de este enfoque clásico al centrarse directamente en el crecimiento del *soporte* de la configuración, es decir, en la tasa a la que el cabezal descubre celdas nuevas de la cinta, y no en el promedio de celdas leídas por unidad de tiempo. Si $s_t(x)$ denota el número de celdas distintas visitadas por el cabezal en los primeros t pasos a partir de la configuración x , se define el *movement bound* de la máquina como

$$m(t) = \max_x s_t(x),$$

cantidad que mide precisamente cuántas celdas nuevas puede conocer, en el peor caso, cualquier configuración de la máquina hasta el tiempo t [7]. A partir de $m(t)$ se define la *velocidad* de la máquina M como el límite

$$\text{speed}(M) = \lim_{t \rightarrow \infty} \frac{m(t)}{t}$$

[7]. Bajo esta definición se establece una tricotomía completa: la máquina tiene movimiento acotado (y por tanto es preperiódica), o bien $m(t) = \Theta(t/\log t)$ y la máquina es *distorsionada* (aperiódica con crecimiento sublineal del soporte), o bien $m(t) = \Theta(t)$ y la máquina tiene velocidad positiva [7]. En particular, se demuestra que toda máquina de un cabezal aperiódica es necesariamente distorsionada [7]. Es precisamente esta noción de $m(t)$, y no la definición basada en promedio asintótico, la que se extiende en esta tesis para binSMART, al determinar el tiempo mínimo $Q(k)$ necesario para descubrir k celdas nuevas y establecer que $m(t) = \Theta(\log_2 t)$. Complementariamente, se probó que decidir la *positividad exacta* de la velocidad, en el sentido de la definición clásica basada en promedio asintótico, es indecidible en máquinas reversibles y completas [15].

2.1.4 Las máquinas SMART y binSMART

La máquina SMART está definida sobre un alfabeto de tres símbolos $\{0, 1, 2\}$ y posee cuatro estados internos, denotados b , d , p y q [3]. Su diagrama de transiciones (Figura 2.1) muestra una estructura simétrica: los estados b y d escriben el mismo símbolo sobre la cinta pero se desplazan en direcciones opuestas, y ocurre lo análogo entre p y q [3]. Cada arco del diagrama, etiquetado de la forma $s|s'\triangleright$ o $s|s'\triangleleft$, representa una instrucción que lee el símbolo s , escribe s' y mueve el cabezal a la derecha ($\triangleright$) o a la izquierda ($\triangleleft$). El estado p actúa como nodo central, conectado bidireccionalmente con q mediante la instrucción $2|0$, y unidireccionalmente hacia b (instrucciones $1|1$ y $2|2$) y desde d (instrucciones $0|2$ y $1|0$). Los estados b y d están además conectados entre sí por un par de arcos largos etiquetados $0|1$,

que constituyen el mecanismo por el cual la máquina invierte su búsqueda al alcanzar el extremo de un bloque de ceros. Esta estructura de cuatro estados es la que permite que SMART sea, simultáneamente, completa, reversible, aperiódica, simétrica en el tiempo y topológicamente minimal, siendo hasta la fecha el ejemplo más pequeño conocido con esta combinación de propiedades [3].

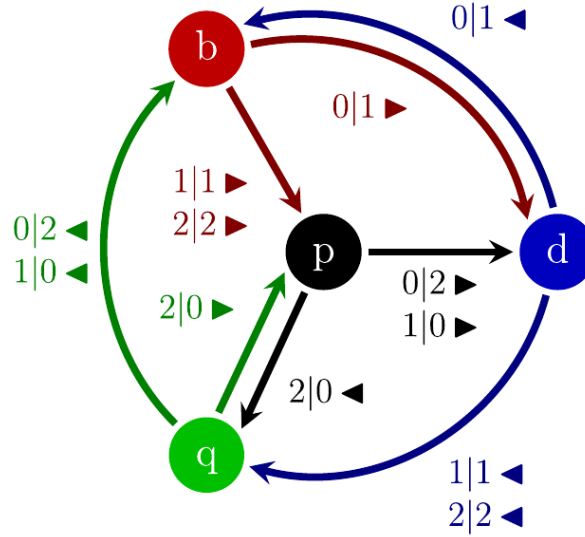


Figura 2.1: Diagrama de estados y transiciones de la máquina SMART, con cuatro estados (b, d, p, q) sobre el alfabeto $\{0, 1, 2\}$.

La máquina BINSMART, propuesta en [4], es una reconstrucción de SMART sobre un alfabeto binario $\{0, 1\}$, lo que obliga a duplicar el número de estados internos para preservar la información que en SMART codificaba el tercer símbolo [4]. Su diagrama de transiciones (Figura 2.2) posee ocho estados, denotados $1D, 1D', 1G, 1G', 3D, 3D', 3G$ y $3G'$, organizados en dos familias simétricas: los estados con sufijo D y D' replican el comportamiento de búsqueda hacia la derecha, mientras que los estados con sufijo G y G' replican el comportamiento simétrico hacia la izquierda. Los pares $(1D, 1D')$ y $(3D, 3D')$ —y análogamente $(1G, 1G')$ y $(3G, 3G')$ — corresponden a las dos etapas necesarias para codificar en binario cada movimiento que en SMART se ejecutaba con un solo estado y un símbolo ternario. A diferencia de SMART, esta reconstrucción *no* preserva la simetría temporal de la máquina original, ya que no existe un par de involuciones h_Q y h_Σ que conjuguen la máquina con su inversa [4]. No obstante, en [4] se demuestra que BINSMART conserva las propiedades de completitud, reversibilidad, aperiodicidad y minimalidad topológica de SMART, y que su t-shift es una subshift sustitutiva, generada por una función recursiva de trazas [4]. La demostración de aperiodicidad y minimalidad se apoya en un conjunto de proposiciones de movimiento básico —denotadas $D_1(n), G_1(n), D'_1(n), G'_1(n), D_3(n), G_3(n), D'_3(n)$ y $G'_3(n)$ — cuya validez para todo $n \in \mathbb{N}$ se establece mediante inducción en el **Lemma 1** de [4]: cada movimiento de escala

n se descompone recursivamente en movimientos de escala $n - 1$, de forma análoga a como los estados de la Figura 2.2 se conectan en cadenas que amplían progresivamente el bloque de ceros explorado. Este lema constituye la base formal sobre la cual se construye la *toolbox* de movimientos utilizada en la presente tesis, dado que cada operación compuesta sobre la cinta de BINSMART puede expresarse como una composición finita de estos movimientos elementales certificados por el Lemma 1 [4].

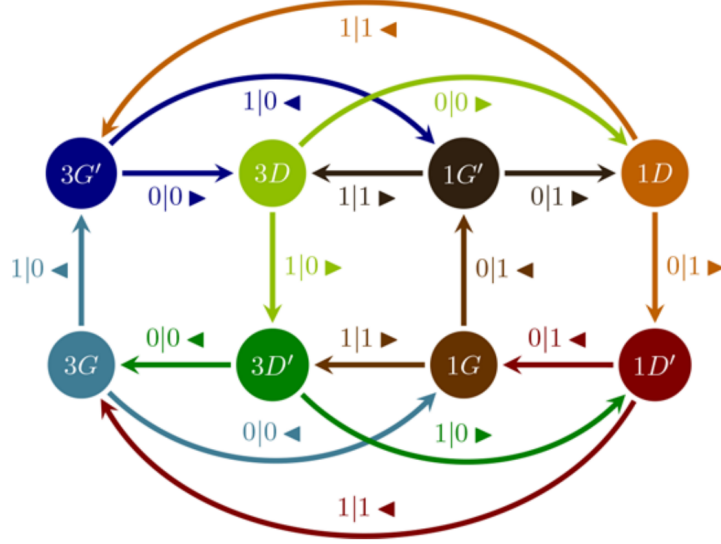


Figura 2.2: Diagrama de estados y transiciones de la máquina binSMART, con ocho estados sobre el alfabeto binario $\{0, 1\}$.

2.1.5 Comportamiento de ida y vuelta (zigzag)

Tanto SMART como BINSMART recorren la cinta mediante un patrón de *ida* y *vuelta* acotado: el cabezal se desplaza en una dirección hasta encontrar un símbolo distintivo, retrocede, y repite el proceso ampliando progresivamente el rango de búsqueda antes de continuar en una nueva dirección [3, 4]. Este comportamiento recuerda formalmente a lo que en [5] se denomina una *zigzag*: una máquina realiza un *right-zigzag* (o *left-zigzag*) de anchura N sobre una celda i si existen instantes $t_0 < t_1 < t_2$ tales que el cabezal está en la posición i en t_0 y t_2 , y en $i + N$ (o $i - N$) en t_1 [5]. Una máquina se dice *bounded-zigzag* si la anchura máxima de sus zigzags es finita, propiedad que en [5] se vincula directamente con la complejidad del lenguaje formal del t-shift asociado: se demostró que una máquina es *bounded-zigzag* si y solo si su t-shift S_T es reconocido por un autómata de pila determinista (DPDA), y que si el t-shift es sofic, la máquina es necesariamente de *bounded-cycle* [5].

Es importante señalar, sin embargo, que el comportamiento de ida y vuelta de SMART y BINSMART **no** corresponde al caso *bounded-zigzag* estudiado en [5], sino a su contraparte de anchura no aco-

tada. En el marco de [5], la anchura N de cada zigzag está sujeta a una cota fija, lo que permite codificar la información necesaria para recordar.^{el} punto de retorno mediante una pila —de ahí la caracterización vía DPDA—. En BINS_{MART}, en cambio, el símbolo distintivo que marca cada punto de retorno (el símbolo 1 en los movimientos elementales del Lema 6.1 de [4]) se desplaza progresivamente hacia el infinito a medida que avanza la computación: cada búsqueda de ida y vuelta tiene una anchura n estrictamente mayor que la anterior, de modo que la sucesión de anchuras $\{n\}_{n \geq 0}$ no está acotada. Por tanto, BINS_{MART} no es bounded-zigzag, y el resultado de equivalencia con DPDA no se aplica directamente a esta máquina; el marco de [5] es útil aquí únicamente como vocabulario descriptivo para nombrar el patrón de movimiento —zigzag de anchura creciente—, y no como herramienta de clasificación de complejidad del t-shift asociado a BINS_{MART}.

2.1.6 Weak Mixing

El *weak mixing* (mezcla débil) es una propiedad dinámica más fuerte que la transitividad topológica, definida mediante la condición de que el sistema producto $(X \times X, T \times T)$ sea transitivo [14]. Esta noción se ubica en una jerarquía estricta: la mezcla topológica implica mezcla débil, y esta implica transitividad total, sin que ninguna de estas implicaciones sea reversible en sistemas de complejidad suficiente [14]. En [14] se demostró en 2022 que la máquina SMART es topológicamente débilmente mezclante en su modelo TMT —y por tanto totalmente transitiva—, y que la mezcla débil, la mezcla topológica y la transitividad total son en general indecidibles para máquinas de Turing, utilizando la técnica de *embedding* que inserta una máquina invitada inocua dentro de una máquina anfitriona [14]. En el contexto más general de autómatas celulares, en [12] se demostró que tanto la mezcla topológica como la sensibilidad a condiciones iniciales son propiedades indecidibles incluso para autómatas celulares reversibles unidimensionales [12], un resultado estructuralmente análogo al obtenido para máquinas de Turing.

2.1.7 Universalidad física

Un eje distinto de investigación explora el extremo opuesto de controlabilidad dinámica: la *universalidad física*. En [13] se define una máquina de Turing como físicamente universal si, para cualquier región finita de la cinta y cualquier transformación deseada sobre su contenido, existe una configuración del resto de la cinta y un tiempo de evolución que implementan dicha transformación exactamente [13]. Allí se construye explícitamente una máquina bidimensional de cinco estados y radio uno que es eficientemente físicamente universal tanto en el modelo de cinta móvil como en el de cabezal móvil, demostrando además que es mixing de todo orden finito en ambos modelos y que carece de órbitas periódicas en el modelo de cabezal móvil, condición necesaria para la universalidad física [13]. Este resultado contrasta con el comportamiento de SMART y BINS_{MART}: mientras estas últimas son minimales y aperiódicas pero exhiben un patrón de movimiento altamente estructurado y predecible (las búsquedas acotadas descritas

por el Lemma 1 y los zigzags de [5]), una máquina físicamente universal debe ser capaz de generar, a partir de su propia dinámica interna, cualquier transformación computable sobre una región de la cinta, lo que la sitúa en el extremo de mayor control dinámico posible dentro de la jerarquía de propiedades revisada en este estado del arte [13].

Este marco conceptual —que combina la caracterización TMT/TMH [11], la construcción y el Lemma 1 de la toolbox de BINSMART [4], el análisis de zigzags [5], las técnicas de *embedding* aplicadas sobre SMART [3, 14], y el contraste con la universalidad física [13]— constituye la base metodológica sobre la cual se sitúa el estudio de la dinámica de la máquina BINSMART y su toolbox asociada.

2.2 Discusión

La construcción de la toolbox de SMART como metodología a replicar

El antecedente metodológico más directo para la toolbox desarrollada en esta tesis no proviene de [3], sino del trabajo posterior sobre las nociones de mezcla topológica en SMART [14]. Mientras que la prueba original de aperiodicidad y minimalidad de SMART en [3] se apoya en un *Lemma 1* puramente cualitativo —las proposiciones B_n , D_n , P_n y Q_n afirman únicamente que la máquina *eventualmente* atraviesa cualquier laguna de ceros de longitud n , sin precisar cuántos pasos requiere para hacerlo—, la demostración de que SMART es débilmente mezclante exige una cota temporal explícita: para sincronizar dos configuraciones distintas dentro de una ventana de tiempo común, no basta saber que un movimiento ocurre, es necesario saber *exactamente cuántos pasos* tarda en ocurrir [14].

Por este motivo, en la Sección 3 de [14] —titulada explícitamente “SMART toolbox”— se reescriben los movimientos básicos de SMART dotándolos de fórmulas cerradas para su duración exacta. El *Lemma 2* de [14] establece, por inducción sobre n , que atravesar una laguna de n ceros desde los estados b , d , p o q toma exactamente

$$Z_n = 3Z_{n-1} + 3 = \frac{3}{2}(3^{n+1} - 1)$$

pasos, con caso base $Z_0 = 3$ verificado por simulación directa. La demostración replica exactamente la estructura recursiva del Lemma 1 de [3] —cada movimiento de escala n se descompone en tres llamadas a movimientos de escala $n - 1$ (un *Hooper call* anidado)— pero en cada paso de la inducción se *suman* los tiempos parciales en lugar de solo verificar la existencia de la transición. A partir de Z_n , el *Lemma 3* deriva el tiempo intermedio $O_n = 3^{n+1}$ correspondiente a una versión truncada del mismo movimiento; el *Lemma 4* cuantifica, mediante $T_{n,i} = 3^{n+2}(8 \cdot 3^{2i} - 1)$, el tiempo que toma expandir progresivamente el tamaño de la laguna explorada; y el *Lemma 5* cuantifica, mediante $K_{n,i} = 5 \cdot 3^{n+1}(8 \cdot 3^{2i} - 1)$, el tiempo que toma rellenar la cinta con símbolos positivos tras una búsqueda. Cada una de estas cuatro fórmulas se obtiene por inducción sobre el parámetro de escala (n o i), reduciendo siempre el movimiento compuesto a una combinación aritmética de invocaciones ya cuantificadas de escala menor —exactamente el mismo

principio de descomposición recursiva que sostiene el Lemma 1 de la toolbox de binSMART en [4], solo que aquí cada paso de la recursión lleva adosado su costo temporal exacto en lugar de solo su validez cualitativa.

El paper original de binSMART y su Lemma 1

El segundo texto relevante para esta discusión corresponde al artículo original de [4], “A Binary Complete and Aperiodic Turing Machine”, publicado en el *International Journal of Unconventional Computing* en 2020. Este trabajo introduce la máquina binSMART como una reconstrucción de dos símbolos de la dinámica principal de la máquina SMART de [3], y demuestra para ella las propiedades de aperiodicidad, minimalidad topológica y transitividad, además de probar que su t-shift es una sustitución primitiva.

En la Sección 3.2 de ese artículo (“BinSmart’s behavior”), se describen el comportamiento de la máquina mediante ocho proposiciones de movimiento acotado, definidas textualmente como

$$\begin{aligned}
D_1(n) : \begin{pmatrix} 0 & 0^n & 1 \\ 1D \end{pmatrix} \vdash^* \begin{pmatrix} 1 & 0^n & 1 \\ 3G \end{pmatrix}, \quad G_1(n) : \begin{pmatrix} 1 & 0^n & 0 \\ 1G \end{pmatrix} \vdash^* \begin{pmatrix} 1 & 0^n & 1 \\ 3D \end{pmatrix} \\
D'_1(n) : \begin{pmatrix} 1 & 0 & 0^n & s \\ 1D' \end{pmatrix} \vdash^* \begin{pmatrix} 1 & 0 & 0^n & s \\ 1D' \end{pmatrix}, \quad G'_1(n) : \begin{pmatrix} s & 0^n & 0 & 1 \\ 1G' \end{pmatrix} \vdash^* \begin{pmatrix} s & 0^n & 0 & 1 \\ 1G' \end{pmatrix} \\
D_3(n) : \begin{pmatrix} 0 & 0^n & 1 \\ 3D \end{pmatrix} \vdash^* \begin{pmatrix} 0 & 0^n & 1 \\ 3D \end{pmatrix}, \quad G_3(n) : \begin{pmatrix} 1 & 0^n & 0 \\ 3G \end{pmatrix} \vdash^* \begin{pmatrix} 1 & 0^n & 0 \\ 3G \end{pmatrix} \\
D'_3(n) : \begin{pmatrix} 1 & 0^n & 0 \\ 3D' \end{pmatrix} \vdash^* \begin{pmatrix} 1 & 0^n & 0 \\ 3G \end{pmatrix}, \quad G'_3(n) : \begin{pmatrix} 0 & 0^n & 1 \\ 3G' \end{pmatrix} \vdash^* \begin{pmatrix} 0 & 0^n & 1 \\ 3D \end{pmatrix}
\end{aligned}$$

y se enuncia el **Lemma 1** bajo la afirmación única de que “ $D_1(n)$, $G_1(n)$, $D'_1(n)$, $G'_1(n)$, $D_3(n)$, $G_3(n)$, $D'_3(n)$ y $G'_3(n)$ son verdaderas para todo $n \in \mathbb{N}$ ”, demostrada por inducción sobre n , descomponiendo cada movimiento de escala n en llamadas anidadas a movimientos de escala $n - 1$ y $n - 2$ [4]. La demostración es puramente existencial: en cada paso inductivo se verifica que el cabezal efectivamente transita de la configuración de la izquierda a la de la derecha, pero en ningún momento se registra ni se suma el número de pasos que toma dicha transición [4].

Este Lemma 1 es, en definitiva, el que define el comportamiento estructural de binSMART —las búsquedas acotadas de unos que dan nombre a la Sección 3.2— y sobre el cual se construye el resto del artículo: la aperiodicidad (Lemmas 2 a 4 y Teorema 1), la no simetría temporal (Proposición 1) y la minimalidad topológica (Lemmas 5 a 10 y Teorema 2) se apoyan en la existencia de estos movimientos, pero ninguna de esas demostraciones requiere ni aporta una cuantificación del tiempo que toman [4]. Es precisamente esta laguna —una definición del comportamiento sin cuantificación asociada— la que

la construcción de la toolbox de esta tesis busca cerrar, dotando a cada una de estas ocho proposiciones de una función de tiempo cerrada y demostrada por inducción, siguiendo el mismo procedimiento metodológico que [14] aplicó posteriormente sobre SMART.

El paper de Kůrka y el origen de la conjetura

El tercer texto relevante para esta discusión es el trabajo fundacional de [11], “On topological dynamics of Turing machines”, publicado en *Theoretical Computer Science* en 1997. Este artículo formaliza el modelo de *Turing machine with moving tape* (TMT) que tanto SMART como binSMART adoptan como marco dinámico, definiendo el espacio de configuraciones $X = Q \times A^{\mathbb{Z}}$ como un sistema dinámico topológico compacto, sin necesidad de estados inicial o final [11].

En [11] se demuestra que ninguna TMT tiene puntos periódicos atractores (Proposición 5) y se caracteriza la propiedad de *shadowing* en términos de la función de posición relativa $\alpha(x, t)$, que mide el desplazamiento acumulado de la cabeza hasta el instante t [11]. Sobre esta base, se formula la **Conjetura 1**: “every TMT has a periodic point” [11], es decir, que toda máquina de Turing con cinta móvil posee al menos una configuración periódica. Esta conjetura es puramente cualitativa y estructural: no involucra ninguna cota temporal, sino únicamente la existencia de un punto fijo bajo alguna iteración de la dinámica.

La conjetura de Kůrka fue refutada años más tarde, primero por una máquina irreversible [2], y posteriormente de manera definitiva por SMART [3], que además demostró ser reversible, completa y topológicamente minimal —propiedades no vistas antes en el marco TMT—. binSMART [4] extiende este resultado a un alfabeto binario, preservando la aperiodicidad y la minimalidad topológica, aunque perdiendo la simetría temporal de SMART. En ambos casos, la demostración de aperiodicidad recurre exactamente a la maquinaria formal introducida en [11]: la función de posición $\alpha(x, t)$ y el análisis de retornos acotados (Lemma 3 y Teorema 6 de [11]) son el antecedente directo de los Lemmas 2 a 4 de binSMART, donde se prueba que ninguna configuración vuelve a sí misma bajo la dinámica [4].

Es importante notar que ni la conjetura original de [11] ni su refutación en SMART o binSMART requieren cuantificar el número de pasos que toma cada movimiento —el marco de [11] trabaja con la existencia de límites de la función de posición ($\lim \alpha(x, t) = \pm\infty$), no con su valor exacto [11]—. Esto refuerza el punto central de esta tesis: la construcción de la toolbox cuantitativa de binSMART no solo cierra la laguna dejada por el Lemma 1 de [4], sino que además aporta un tipo de información —el tiempo exacto de cada movimiento elemental— que el marco topológico de [11] nunca necesitó ni proveyó, abriendo una vía cuantitativa sobre un problema que históricamente se ha estudiado solo de manera cualitativa.

El antecedente del *speed positiveness problem*

Un cuarto texto relevante para esta discusión es el trabajo de [15], “Undecidability of the speed positiveness problem in reversible and complete Turing machines”, publicado en el *Journal of Computer and System Sciences* en 2021. Este artículo define la velocidad de una configuración c como el límite asintótico $\bar{s}(c) = \limsup_{n \rightarrow \infty} s_n(c)/n$, donde $s_n(c)$ es el número de celdas distintas visitadas hasta el tiempo n , y demuestra que decidir si una máquina de Turing reversible y completa arbitraria admite alguna configuración con velocidad positiva es, en general, indecidible [15].

Es importante precisar que esta noción de velocidad no es idéntica a la cantidad que se cuantifica en esta tesis: mientras que $\bar{s}(c)$ es un límite asintótico que solo distingue si la velocidad es positiva o nula, el objeto de estudio de esta tesis es el número exacto de pasos $Q(k)$ requerido para descubrir k celdas nuevas, para cada valor finito de k . Ambas nociones están relacionadas: una cota lineal exacta como $Q(k) = \Theta(k)$ implica, en particular, que la configuración correspondiente tiene velocidad positiva en el sentido de [15]. Sin embargo, el resultado de indecidibilidad de [15] se refiere a la pregunta binaria (positiva o no) aplicada a toda la clase de máquinas reversibles y completas, mientras que esta tesis se restringe a un análisis exacto y explícito de un caso particular, binSMART, motivado precisamente por el hecho de que ninguna respuesta general puede heredarse automáticamente de un resultado aplicable a toda la clase [15].

Capítulo 3

Definición del problema

A pesar de que la máquina SMART ha sido extensamente caracterizada —incluyendo la demostración de que es topológicamente débilmente mezclante (*weak mixing*) en su modelo TMT [14]—, no se conoce si su reconstrucción binaria BINS_{MART} conserva esta propiedad [4]. La principal dificultad para responder esta pregunta radica en que, hasta ahora, no se contaba con una *toolbox* de movimientos elementales suficientemente desarrollada para BINS_{MART} que permitiera construir y componer las trayectorias necesarias para verificar (o refutar) la condición de transitividad del sistema producto $(X \times X, T \times T)$ exigida por la definición de weak mixing [14]. Si bien el Lemma 1 [4] establece la validez de los movimientos básicos $D_1(n)$, $G_1(n)$, $D_3(n)$ y $G_3(n)$ para todo $n \in \mathbb{N}$, dicho resultado no cuantifica el costo temporal de estos movimientos ni permite, por sí solo, determinar cuál es la configuración que recorre la cinta en el menor número de pasos posible.

En consecuencia, el problema abordado en esta tesis es doble. Primero, no se dispone de una *toolbox* de movimientos con complejidad temporal explícita que permita operar formalmente sobre BINS_{MART} para estudiar propiedades dinámicas finas como el weak mixing. Segundo, y como paso previo indispensable, no se conoce cuál es la configuración inicial más rápida capaz de descubrir k celdas nuevas de la cinta, ni el tiempo mínimo $Q(k)$ que dicha configuración óptima requiere para lograrlo, dato que resulta esencial para acotar cualquier argumento de transitividad o mezcla sobre el sistema.

3.1 Alcance

El alcance de esta tesis se limita a dos contribuciones concretas, concebidas como la base necesaria para atacar en trabajo futuro el problema de weak mixing de BINS_{MART}:

- **Construcción de la toolbox de movimientos.** A partir del Lemma 1 de [4], se construye una toolbox de movimientos elementales para BINS_{MART} ($D_1(n)$, $G_1(n)$, $D'_1(n)$, $G'_1(n)$, $D_3(n)$, $G_3(n)$, $D'_3(n)$, $G'_3(n)$), estableciendo y demostrando por inducción sus tiempos de ejecución exactos mediante relaciones de recurrencia (Lemmas 1–3 del análisis desarrollado en esta tesis), lo que permite expresar cualquier movimiento compuesto sobre la cinta como una composición finita de

movimientos con costo temporal conocido.

- **Estudio de la dinámica de velocidad de la configuración más rápida.** Utilizando la toolbox construida, se determina el tiempo mínimo $Q(k)$ necesario para que alguna configuración inicial de BINSMART descubra k celdas nuevas de la cinta sin retroceder, identificando explícitamente las familias de configuraciones óptimas que alcanzan dicho tiempo. Se extiende además este resultado al caso general de trayectorias *unbounded* (Definición 1 y Teorema 1 de esta tesis), demostrando que ninguna trayectoria que cambie de dirección puede igualar el tiempo de la configuración óptima unidireccional, estableciendo así que $Q(k)$ es efectivamente el tiempo mínimo global para descubrir k celdas nuevas, independientemente de la dirección de exploración.

Quedan explícitamente fuera del alcance de esta tesis la demostración (o refutación) de que BINSMART es *weak mixing*, así como cualquier otra propiedad de mezcla o transitividad total del sistema; estos resultados de velocidad y la toolbox construida constituyen únicamente el fundamento sobre el cual dicha pregunta podría abordarse en investigaciones futuras. Cabe señalar que existe además una vía alternativa, de naturaleza puramente algebraica, para abordar esta pregunta: dado que el *t-shift* de BINSMART es la clausura del punto fijo de una sustitución primitiva σ (Teorema 3 [4]), es posible estudiar la propiedad de weak mixing a través del análisis espectral de la matriz de sustitución asociada a σ —en particular, examinando si sus valores propios distintos del valor propio de Perron-Frobenius son o no raíces de la unidad—, sin requerir en absoluto un análisis cuantitativo de tiempos de recorrido como el desarrollado en esta tesis. Esta vía queda igualmente fuera del alcance del presente trabajo, pero se menciona como un camino complementario e independiente que investigaciones futuras podrían explorar en paralelo al enfoque cuantitativo aquí propuesto.

Capítulo 4

Objetivos e hipótesis

4.1 Objetivos de la Investigación

4.1.1 Objetivo General

Construir una *toolbox* analítica y cuantitativa para la máquina de Turing binSMART, que permita describir, formalizar y demostrar con precisión su comportamiento dinámico.

4.1.2 Objetivos Específicos

1. Estudiar y comprender las propiedades dinámicas y topológicas de la máquina SMART, estableciendo las bases conceptuales para el análisis de binSMART.
2. Formalizar las funciones de paso para las operaciones fundamentales de binSMART, expresando el costo temporal de cada patrón de comportamiento mediante fórmulas cerradas.
3. Realizar simulaciones computacionales de la evolución de binSMART sobre diversas configuraciones iniciales, registrando trayectorias y tiempos de cómputo.
4. Identificar y caracterizar patrones de comportamiento a partir de las simulaciones, en particular las configuraciones que minimizan el número de pasos para descubrir k celdas nuevas.
5. Demostrar matemáticamente los patrones obtenidos en las simulaciones, estableciendo cotas inferiores y superiores para el tiempo de exploración de k celdas bajo cualquier configuración inicial posible.

4.2 Hipótesis

Se hipotetiza que es posible construir una *toolbox* analítica de movimientos elementales para binSMART —formalizando los patrones recursivos de búsqueda descritos en el Lemma 1 [4] mediante funciones de paso con costo temporal cerrado— que resulte suficiente para describir con precisión cualquier trayectoria

de la máquina sobre la cinta, sin necesidad de recurrir a simulación exhaustiva. Se espera que dicha toolbox, al sistematizar lemas auxiliares sobre el costo de recorrer segmentos de cinta y de retornar sobre posiciones ya visitadas, junto con los patrones de inducción propios de binSMART, permita derivar de manera rigurosa y reproducible cotas exactas sobre el comportamiento dinámico de la máquina.

En particular, se conjetura que existe una configuración inicial (o familia de configuraciones) que resulta óptima en el sentido de minimizar el número de pasos necesarios para descubrir k celdas nuevas de la cinta, y que dicho tiempo mínimo puede expresarse mediante una fórmula cerrada dependiente de k . Asimismo, se hipotetiza que ninguna otra configuración inicial —incluyendo aquellas cuya trayectoria recorre celdas en ambas direcciones (unbounded)— puede igualar dicho tiempo mínimo, de modo que la configuración óptima identificada sea efectivamente óptima a nivel global y no solo dentro de una clase restringida de trayectorias. Se espera, además, que el marco analítico desarrollado sea extensible al estudio de propiedades dinámicas análogas en otras máquinas de Turing binarias, más allá del caso particular de la velocidad de exploración abordado en esta tesis.

Capítulo 5

Metodología

La metodología de esta investigación se organiza en torno a los cinco objetivos específicos planteados en el capítulo anterior, junto con una planificación temporal que distribuye las tareas a lo largo del desarrollo del proyecto. La tesis se inició en 2025, y la planificación original proyectaba su cierre dentro del segundo semestre de ese mismo año académico.

En una primera etapa (primer semestre de 2025), se cumplió el **Objetivo 1**, consistente en el estudio y comprensión de las propiedades dinámicas y topológicas de la máquina SMART, así como de los trabajos previos asociados (Kůrka [11], Cassaigne, Ollinger y Torres-Avilés [3], Concha-Vega y Torres-Avilés [4], entre otros). Este proceso permitió consolidar la base conceptual y técnica necesaria —nociones de aperiodicidad, minimalidad, reversibilidad y t-shift— para extender dichas ideas al caso binario de binSMART. Esta fase inicial tomó la totalidad del primer semestre, ya que involucró un estudio exhaustivo de las nociones dinámicas, los resultados formales disponibles en la literatura y las técnicas de demostración empleadas, en particular el Lemma 1 [4], que establece la validez de los movimientos básicos de binSMART y constituye el punto de partida de la toolbox construida en esta tesis.

Para el segundo semestre de 2025 se proyectó la realización de los siguientes objetivos:

- **Objetivo 2 (Formalización de las funciones de paso de binSMART):** se destinaron alrededor de 6 semanas a la definición rigurosa de las operaciones elementales de la toolbox ($D_1(n)$, $G_1(n)$, $D_3(n)$, $G_3(n)$, entre otras), demostrando por inducción sobre el modelo TMT las fórmulas cerradas que cuantifican su costo temporal, entre ellas $T(n) = \frac{1}{6}(-(-1)^{n+1} + 2^{n+5} - 15)$, así como las funciones derivadas $M(n)$ y $Z(n)$ que expresan el costo de los movimientos compuestos en términos de $T(n)$. Este objetivo se completó dentro del plazo estimado.
- **Objetivo 3 (Simulaciones computacionales de binSMART):** se destinaron aproximadamente 5 semanas a la implementación de un simulador de la evolución de binSMART sobre el modelo TMT, ejecutando la máquina sobre múltiples configuraciones iniciales —variando el estado inicial $q \in Q$ y el contenido de la cinta— con el fin de registrar trayectorias, tiempos de cómputo y el desplazamiento acumulado $\alpha(x, t)$ de la cabeza en cada paso. Cabe precisar que, bajo el modelo TMT

adoptado en esta tesis, el cabezal permanece fijo en el origen y es la cinta la que se desplaza en cada transición; $\alpha(x, t)$ registra por tanto la posición relativa efectiva de exploración, y no un desplazamiento físico del cabezal. Este objetivo también se completó dentro del plazo estimado, generando la evidencia empírica necesaria para orientar la búsqueda de patrones en la etapa siguiente.

- **Objetivo 4 (Identificación de patrones de comportamiento):** originalmente se había proyectado un período de 4 semanas para este análisis, centrado en el estudio sistemático de las ramas de ejecución posibles según el estado y el símbolo inicial del cabezal, descartando las configuraciones que alcanzan la posición k en un tiempo mayor al mínimo observado. Sin embargo, la exhaustividad requerida para descartar rigurosamente cada rama de ejecución —incluyendo los sub-casos anidados que aparecen al analizar los estados $3D'$, $1G'$ y sus transiciones asociadas— resultó considerablemente más compleja de lo anticipado, extendiendo esta etapa más allá del plazo original.
- **Objetivo 5 (Demostración matemática de los patrones):** de manera similar, la formalización rigurosa de los patrones observados —en particular la demostración de los lemas de costo de segmento y costo de retorno, y su generalización al Teorema 1 sobre trayectorias zigzag— demandó un esfuerzo de demostración mayor al proyectado inicialmente en 5 semanas, dado que fue necesario establecer cotas inferiores válidas para *todas* las configuraciones iniciales posibles, y no solo para las familias óptimas identificadas empíricamente en el Objetivo 4.

En conjunto, los Objetivos 4 y 5 tomaron considerablemente más tiempo que el estipulado originalmente en la planificación del segundo semestre de 2025, extendiéndose el desarrollo de la investigación hasta junio de 2026. Este retraso se explica principalmente por la naturaleza combinatoria del análisis de ramas requerido para garantizar que $Q(k)$ correspondiera efectivamente al tiempo mínimo global —y no solo a un mínimo local observado en simulación—, así como por la necesidad de formalizar el caso de trayectorias zigzag mediante los lemas auxiliares de costo de segmento y costo de retorno, cuya demostración por inducción requirió múltiples iteraciones de refinamiento antes de alcanzar una prueba completa y correcta. A pesar de esta extensión de plazos, la planificación permitió que cada etapa contara finalmente con el tiempo necesario para avanzar desde la formalización teórica de la toolbox (Objetivo 2), pasando por la exploración computacional (Objetivo 3) y la caracterización empírica de patrones (Objetivo 4), hasta la validación matemática rigurosa de dichos patrones (Objetivo 5), de modo que las conclusiones del estudio —en particular la optimalidad global de $Q(k)$ — no dependieran únicamente de la evidencia empírica obtenida en simulación.

Capítulo 6

Resultados

6.1 Construcción de la toolbox de binSMART

En esta sección se construye la toolbox de movimientos elementales de la máquina BINSMART, extendiendo el Lemma 1 [4] mediante el cálculo explícito del costo temporal de cada patrón de comportamiento. El objetivo es expresar, para cada movimiento básico, una fórmula cerrada que cuantifique el número exacto de pasos que la máquina requiere para ejecutarlo, de modo que estas fórmulas puedan combinarse posteriormente para analizar trayectorias arbitrarias sobre la cinta.

6.1.1 Movimientos de escala n : D'_1 y D_3

Lema 6.1. *Consideremos*

$$T(n) = \frac{1}{6} \left(-(-1)^{n+1} + 2^{n+5} - 15 \right).$$

Entonces, para binSMART y para todo $n \geq 0$, se cumple:

$$\begin{aligned} D'_1 : (\forall s \in \{0, 1\}) & \begin{pmatrix} 1 & 0 & 0^n & s \\ & 1D' & & \end{pmatrix} \vdash^{T(n)} \begin{pmatrix} 1 & 0 & 0^n & s \\ & 1D' & & \end{pmatrix} \\ G'_1 : (\forall s \in \{0, 1\}) & \begin{pmatrix} s & 0^n & 0 & 1 \\ & & 1G' & \end{pmatrix} \vdash^{T(n)} \begin{pmatrix} s & 0^n & 0 & 1 \\ 1G' & & & \end{pmatrix} \\ D_3 : & \begin{pmatrix} 0 & 0^n & 1 \\ 3D & & \end{pmatrix} \vdash^{T(n)} \begin{pmatrix} 0 & 0^n & 1 \\ & 3D & \end{pmatrix} \\ G_3 : & \begin{pmatrix} 1 & 0^n & 0 \\ & 3G & \end{pmatrix} \vdash^{T(n)} \begin{pmatrix} 1 & 0^n & 0 \\ 3G & & \end{pmatrix} \end{aligned}$$

Demostración. Hacemos una inducción sobre n . Solamente demostramos D'_1 y D_3 , ya que G'_1 y G_3 son simétricas por reflexión espacial.

Base de inducción $n = 0$.

Para $D_3(0)$

$$\begin{aligned} \begin{pmatrix} 0 & 1 \\ 3D \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 1 \\ 1D \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 1 \\ 3G' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 1 \\ 3D \end{pmatrix} &\blacksquare \end{aligned}$$

Para $D'_1(0)$

$$\begin{aligned} \begin{pmatrix} 1 & 0 & s \\ 1D' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 1 & s \\ 1G \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 1 & s \\ 3D' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 0 & s \\ 1D' \end{pmatrix} &\blacksquare \end{aligned}$$

Ambas pruebas toman $T(0) = \frac{1}{6}(-(-1) + 2^5 - 15) = 3$ pasos, y por simetría espacial, también $G'_1(0)$ y $G_3(0)$.

Base de inducción $n = 1$.

Para $D_3(1)$

$$\begin{aligned} \begin{pmatrix} 0 & 0 & 1 \\ 3D \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 0 & 1 \\ 1D \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 1 & 1 \\ 1D' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 1 & 1 \\ 3G \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 0 & 1 \\ 3G' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 0 & 1 \\ 3D \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 0 & 1 \\ 1D \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 0 & 1 \\ 3G' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 0 & 0 & 1 \\ 3D \end{pmatrix} &\blacksquare \end{aligned}$$

Para $D'_1(1)$

$$\begin{aligned} \begin{pmatrix} 1 & 0 & 0 & s \\ 1D' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 1 & 0 & s \\ 1G \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 1 & 0 & s \\ 3D' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 0 & 1 & s \\ 1D' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 0 & 1 & s \\ 1G \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 1 & 1 & s \\ 1G' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 1 & 1 & s \\ 3D \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 0 & 1 & s \\ 3D' \end{pmatrix} &\vdash \text{Un paso} \\ \begin{pmatrix} 1 & 0 & 0 & s \\ 1D' \end{pmatrix} &\blacksquare \end{aligned}$$

Ambas pruebas toman $T(1) = \frac{1}{6}(-(-1)^2 + 2^6 - 15) = 8$ pasos, y por simetría espacial, también $G'_1(1)$ y $G_3(1)$.

Paso inductivo. Supongamos que el resultado es verdadero para $n-1$ y $n-2$, y probemos para n .

Para D_3 :

$$\begin{aligned} & \left(\begin{array}{cccccc} 0 & 0 & 0 & 0^{n-3} & 0 & 1 \\ 3D & & & & & \end{array} \right) \vdash \text{Dos pasos} \\ & \left(\begin{array}{cccccc} 0 & 1 & 0 & 0^{n-3} & 0 & 1 \\ & 1D' & & & & \end{array} \right) \vdash D'_1(n-2) \\ & \left(\begin{array}{cccccc} 0 & 1 & 0 & 0^{n-3} & 0 & 1 \\ & & & & 1D' & \end{array} \right) \vdash \text{Un paso} \\ & \left(\begin{array}{cccccc} 0 & 1 & 0 & 0^{n-3} & 0 & 1 \\ & & & & 3G' & \end{array} \right) \vdash G_3(n-2) \\ & \left(\begin{array}{cccccc} 0 & 1 & 0 & 0^{n-3} & 0 & 1 \\ & 3G' & & & & \end{array} \right) \vdash \text{Dos pasos} \\ & \left(\begin{array}{cccccc} 0 & 0 & 0 & 0^{n-3} & 0 & 1 \\ & 3D & & & & \end{array} \right) \vdash D_3(n-1) \\ & \left(\begin{array}{cccccc} 0 & 0 & 0 & 0^{n-3} & 0 & 1 \\ & & & & 3D & \end{array} \right) \end{aligned}$$

Para D'_1 :

$$\begin{aligned} & \left(\begin{array}{cccccc} 1 & 0 & 0^{n-3} & 0 & 0 & 0 & s \\ 1D' & & & & & & \end{array} \right) \vdash D'_1(n-1) \\ & \left(\begin{array}{cccccc} 1 & 0 & 0^{n-3} & 0 & 0 & 0 & s \\ & & & 1D' & & & \end{array} \right) \vdash \text{Dos pasos} \\ & \left(\begin{array}{cccccc} 1 & 0 & 0^{n-3} & 0 & 1 & 1 & s \\ & & & 1G' & & & \end{array} \right) \vdash G'_1(n-2) \\ & \left(\begin{array}{cccccc} 1 & 0 & 0^{n-3} & 0 & 1 & 1 & s \\ 1G' & & & & & & \end{array} \right) \vdash \text{Un paso} \\ & \left(\begin{array}{cccccc} 1 & 0 & 0^{n-3} & 0 & 1 & 1 & s \\ & 3D & & & & & \end{array} \right) \vdash D_3(n-2) \\ & \left(\begin{array}{cccccc} 1 & 0 & 0^{n-3} & 0 & 1 & 1 & s \\ & & & & 3D & & \end{array} \right) \vdash \text{Dos pasos} \\ & \left(\begin{array}{cccccc} 1 & 0 & 0^{n-3} & 0 & 0 & 0 & s \\ & & & & & 1D' & \end{array} \right) \end{aligned}$$

Por hipótesis de inducción, cada subcomputación ya demostrada puede reemplazarse por su tiem-

po correspondiente:

$$D'_1(n-1), D_3(n-1) \rightsquigarrow T(n-1), \quad D'_1(n-2), G'_1(n-2), D_3(n-2), G_3(n-2) \rightsquigarrow T(n-2).$$

En efecto, en las configuraciones intermedias reaparece el mismo patrón ya probado, pero sobre ventanas más pequeñas; por eso se invocan las instancias $n-1$ o $n-2$ de las proposiciones inductivas. Así, al sumar los tiempos de la computación para D'_1 o D_3 , se obtiene la recurrencia:

$$T(n) = T(n-1) + 2T(n-2) + 5.$$

Aplicando la hipótesis de inducción:

$$\begin{aligned} T(n) &= T(n-1) + 2T(n-2) + 5 \\ &= \frac{1}{6}(-(-1)^n + 2^{n+4} - 15) + 2 \cdot \frac{1}{6}(-(-1)^{n-1} + 2^{n+3} - 15) + 5 \\ &= \frac{1}{6}(-(-1)^n + 2^{n+4} - 15 + 2(-1)^n + 2^{n+4} - 30) + 5 \\ &= \frac{1}{6}((-1)^n + 2^{n+5} - 45) + 5 \\ &= \frac{1}{6}((-1)^n + 2^{n+5} - 15). \end{aligned}$$

Finalmente, reescribiendo el signo del término $(-1)^n$ para que coincida con la forma enunciada,

$$T(n) = \frac{1}{6}(-(-1)^{n+1} + 2^{n+5} - 15). \quad \square$$

6.1.2 Movimientos compuestos: D_1 y G_1

Lema 6.2. *Consideremos*

$$M(n) = \frac{1}{3}(-(-1)^n + 2^{n+4} - 9).$$

Entonces, para binSMART y para todo $n \geq 0$, se cumple:

$$\begin{aligned} D_1 : \begin{pmatrix} 0 & 0^n & 1 \\ 1D & & \end{pmatrix} &\vdash^{M(n)} \begin{pmatrix} 1 & 0^n & 1 \\ 3G & & \end{pmatrix} \\ G_1 : \begin{pmatrix} 1 & 0^n & 0 \\ & & 1G \end{pmatrix} &\vdash^{M(n)} \begin{pmatrix} 1 & 0^n & 1 \\ & & 3D \end{pmatrix} \end{aligned}$$

Demostración. Usamos el Lema 6.1 para construir una prueba directa. Demostramos únicamente D_1 , ya

que G_1 es simétrica.

$$\begin{aligned} \left(\begin{array}{ccccc} 0 & 0 & 0^{n-2} & 0 & 1 \\ 1D & & & & \end{array} \right) &\vdash \text{Un paso} \\ \left(\begin{array}{ccccc} 1 & 0 & 0^{n-2} & 0 & 1 \\ & 1D' & & & \end{array} \right) &\vdash D'_1(n-1) \\ \left(\begin{array}{ccccc} 1 & 0 & 0^{n-2} & 0 & 1 \\ & & & 1D' & \end{array} \right) &\vdash \text{Un paso} \\ \left(\begin{array}{ccccc} 1 & 0 & 0^{n-2} & 0 & 1 \\ & & & 3G & \end{array} \right) &\vdash D_3(n-1) \\ \left(\begin{array}{ccccc} 1 & 0 & 0^{n-2} & 0 & 1 \\ 3G & & & & \end{array} \right) & \end{aligned}$$

El costo total resulta de sumar los dos pasos sueltos con las dos invocaciones a movimientos de escala $n-1$ certificados por el Lema 6.1:

$$M(n) = 2T(n-1) + 2 = \frac{1}{3} \left(-(-1)^n + 2^{n+4} - 9 \right). \quad \square$$

6.1.3 Movimientos de transición: D'_3 y G'_3

Lema 6.3. *Consideremos*

$$Z(n) = \frac{1}{6} \left(-(-1)^n + 2^{n+4} - 9 \right).$$

Entonces, para *binSMART* y para todo $n \geq 0$, se cumple:

$$\begin{aligned} D'_3 : \left(\begin{array}{ccc} 1 & 0^n & 0 \\ & & 3D' \end{array} \right) &\vdash^{Z(n)} \left(\begin{array}{ccc} 1 & 0^n & 0 \\ 3G & & \end{array} \right) \\ G'_3 : \left(\begin{array}{ccc} 0 & 0^n & 1 \\ 3G & & \end{array} \right) &\vdash^{Z(n)} \left(\begin{array}{ccc} 0 & 0^n & 1 \\ & & 3D' \end{array} \right) \end{aligned}$$

Demostración. Usamos el Lema 6.1 para construir una prueba directa. Demostramos únicamente D'_3 , ya que G'_3 es simétrica.

$$\begin{aligned} \left(\begin{array}{cccc} 1 & 0^{n-1} & 0 & 0 \\ & & 3D' & \end{array} \right) &\vdash \text{Un paso} \\ \left(\begin{array}{cccc} 1 & 0^{n-1} & 0 & 0 \\ & & 3G & \end{array} \right) &\vdash G_3(n-1) \\ \left(\begin{array}{cccc} 1 & 0^{n-1} & 0 & 0 \\ 3G & & & \end{array} \right) &\blacksquare \end{aligned}$$

El costo total resulta de sumar el paso suelto con la invocación al movimiento de escala $n-1$ certificado

por el Lema 6.1:

$$Z(n) = T(n-1) + 1 = \frac{1}{6} \left(-(-1)^n + 2^{n+4} - 9 \right). \quad \square$$

6.2 Análisis de configuraciones iniciales: velocidad de binSMART

En esta sección aplicamos la toolbox construida en la Sección 6.1 para resolver el problema central de esta tesis: determinar, entre todas las configuraciones iniciales posibles de BINSMART, el número mínimo de pasos necesarios para descubrir k celdas nuevas de la cinta —es decir, celdas que la cabeza no ha visitado previamente—, sin imponer restricción alguna sobre la dirección en que estas celdas se descubren.

Para abordar este problema en toda su generalidad, procedemos por etapas de creciente complejidad. En primer lugar, estudiamos el caso más restringido: trayectorias que descubren las k celdas exclusivamente hacia un lado, digamos hacia la derecha, sin visitar en ningún momento posiciones a la izquierda de la posición inicial. Este caso, más simple de analizar, es también el punto de partida natural, ya que toda trayectoria que descubre celdas en una sola dirección puede reducirse a él. En segundo lugar, extendemos el resultado al caso en que la trayectoria se mueve exclusivamente hacia la derecha o exclusivamente hacia la izquierda. Finalmente, en la etapa más general, consideramos trayectorias que descubren celdas en *ambas* direcciones simultáneamente —trayectorias *unbounded*—, y examinamos si alguna combinación de movimientos hacia la izquierda y hacia la derecha puede igualar el tiempo obtenido por una trayectoria puramente unidireccional. Veremos que el resultado obtenido en la primera etapa no solo resuelve el caso restringido, sino que resulta ser también la referencia natural con la cual comparar el problema general planteado al inicio de esta sección.

Consideramos entonces, para la primera etapa, todas las configuraciones iniciales posibles: cabezal en posición 0, estado $q \in \mathcal{Q}$, sobre una cinta $c \in \{0, 1\}^{k+1}$. El objetivo es determinar cuál configuración alcanza la posición k en el menor número de pasos, sin visitar la posición -1 . Definimos el tiempo mínimo candidato como

$$Q(k) = 3 + T(k-4) = \frac{1}{6} \left(-(-1)^{k-3} + 2^{k+1} + 3 \right), \quad T(n) = \frac{1}{6} \left(-(-1)^{n+1} + 2^{n+5} - 15 \right).$$

Observación (Casos base $k = 1, 2, 3$). Para $k \in \{1, 2, 3\}$, se tiene $Q(k) = k$. En efecto, cada instrucción de binSMART desplaza el cabezal exactamente una posición por paso, de modo que alcanzar la posición k desde la posición inicial 0 requiere al menos k pasos; esta cota se alcanza explícitamente mediante las configuraciones

$$Q(1) = 1 \text{ con } [0, 0]_{1D}, \quad Q(2) = 2 \text{ con } [1, 1, 0]_{1G}, \quad Q(3) = 3 \text{ con } [1, 0, 0, 0]_{1G'}.$$

La fórmula general $Q(k) = 3 + T(k-4)$ solo se demuestra —y solo es necesaria— para $k \geq 4$, rango en

el cual $T(k-4)$ está bien definida ($k-4 \geq 0$, condición exigida en el Lema 6.1).

6.2.1 Estados que alcanzan la posición -1

6.2.1.1 Estado inicial $1D'$

Por las transiciones de binSMART:

$$(1D', 0) \rightarrow (1G, 1, \leftarrow), \quad (1D', 1) \rightarrow (3G, 1, \leftarrow).$$

Para cualquier símbolo $c_0 \in \{0, 1\}$ en posición 0, la cabeza se mueve a la posición -1 en el primer paso.

Toda configuración con estado inicial $1D'$ queda descartada. ■

6.2.1.2 Estado inicial $3G$

Por las transiciones:

$$(3G, 0) \rightarrow (1G, 0, \leftarrow), \quad (3G, 1) \rightarrow (3G', 0, \leftarrow).$$

Para cualquier símbolo $c_0 \in \{0, 1\}$, la cabeza se mueve a la posición -1 en el primer paso.

Toda configuración con estado inicial $3G$ queda descartada. ■

6.2.1.3 Estado inicial $3D'$

Por las transiciones:

$$(3D', 0) \rightarrow (3G, 0, \leftarrow), \quad (3D', 1) \rightarrow (1D', 0, \rightarrow).$$

- Si $c_0 = 0$: la cabeza va a posición -1 en el primer paso. Toda configuración con $c_0 = 0$ queda descartada. ■
- Si $c_0 = 1$: la ejecución comienza escribiendo 0 en la posición 0 y moviendo la cabeza a la posición 1, con estado $1D'$. La configuración inicial tiene entonces la forma

$$\begin{pmatrix} 1 & c_1 & \cdots & c_k \\ 3D' & & & \end{pmatrix},$$

pero, recordando que en la celda original había un 1, tras el primer paso queda

$$\begin{pmatrix} 0 & c_1 & \cdots & c_k \\ 1D' & & & \end{pmatrix}.$$

Por las transiciones de $1D'$,

$$(1D', 0) \rightarrow (1G, 1, \leftarrow), \quad (1D', 1) \rightarrow (3G, 1, \leftarrow),$$

en ambos casos la cabeza se mueve una posición a la izquierda. Por lo tanto, desde la posición 1 la máquina retrocede a la posición 0, ya sea pasando a estado $1G$ o a estado $3G$.

A partir de ahí, la dinámica continúa necesariamente hacia la izquierda:

$$(1G, 0) \rightarrow (1G', 1, \leftarrow), \quad (3G, 0) \rightarrow (1G, 0, \leftarrow),$$

y ambos estados siguen desplazándose hacia la izquierda. En particular, una vez que la máquina entra en $1D'$, no existe una transición que permita avanzar hacia la derecha antes de visitar la posición -1 .

Por consiguiente, toda configuración con $c_0 = 1$ y estado inicial $3D'$ termina alcanzando la posición -1 en tiempo finito. Queda descartada. ■

Toda configuración con estado inicial $3D'$ queda descartada. ■

6.2.1.4 Estado inicial $1G$

Por las transiciones de binSMART,

$$(1G, 0) \rightarrow (1G', 1, \leftarrow), \quad (1G, 1) \rightarrow (3D', 1, \rightarrow).$$

Si $c_0 = 0$, la cabeza alcanza la posición -1 en el primer paso. Por tanto, toda configuración con estado inicial $1G$ y $c_0 = 0$ queda descartada.

Supongamos ahora que $c_0 = 1$. La cabeza avanza a la posición 1 con estado $3D'$. Se distinguen dos casos según c_1 .

- Si $c_1 = 0$, por la transición $(3D', 0) \rightarrow (3G, 0, \leftarrow)$, la cabeza retrocede a la posición 0 con estado $3G$. Como en dicha posición se encuentra el símbolo 1, se aplica $(3G, 1) \rightarrow (3G', 0, \leftarrow)$, y la cabeza alcanza la posición -1 . **Descartada.**
- Si $c_1 = 1$, por la transición $(3D', 1) \rightarrow (1D', 0, \rightarrow)$, la cabeza escribe 0 en la posición 1 y avanza a la posición 2 con estado $1D'$. Las transiciones de $1D'$ hacen que la cabeza retroceda a la posición 1, escribiendo 1 en la posición 2. La configuración queda

$$\begin{pmatrix} 1 & 0 & 1 & c_3 & \cdots & c_k \\ q' & & & & & \end{pmatrix}, \quad q' \in \{1G, 3G\}.$$

Se distinguen dos subcasos según q' .

- Si $q' = 1G$: por la transición $(1G, 0) \rightarrow (1G', 1, \leftarrow)$, la cabeza escribe 1 en la posición 1 y retrocede a la posición 0, obteniendo

$$\begin{pmatrix} 1 & 1 & 1 & c_3 & \cdots c_k \\ 1G' \end{pmatrix}.$$

Esta configuración coincide exactamente con el caso $c_0 = 1, c_1 = 1$ del análisis del **Estado inicial** $1G'$ (Sección 6.2.3.1), donde ya se demostró que dicha rama alcanza la posición -1 en tiempo finito. **Descartada.**

- Si $q' = 3G$: por la transición $(3G, 0) \rightarrow (1G, 0, \leftarrow)$, la cabeza no modifica la posición 1 (permanece en 0) y retrocede a la posición 0, obteniendo

$$\begin{pmatrix} 1 & 0 & 1 & c_3 & \cdots c_k \\ 1G \end{pmatrix}.$$

Esta es exactamente la configuración con estado inicial $1G$ y $c_1 = 0$, ya resuelta al comienzo de esta misma subsección. **Descartada.**

En consecuencia, toda configuración con estado inicial $1G$ queda descartada.

6.2.1.5 Estado inicial $1D$

Por las transiciones

$$(1D, 0) \rightarrow (1D', 1, \rightarrow), \quad (1D, 1) \rightarrow (3G', 1, \leftarrow).$$

Si $c_0 = 1$, por la transición $(1D, 1) \rightarrow (3G', 1, \leftarrow)$, la cabeza escribe 1 en la posición 0 y retrocede a la posición -1 en el primer paso. **Descartada.**

Si $c_0 = 0$, por la transición $(1D, 0) \rightarrow (1D', 1, \rightarrow)$, la cabeza escribe 1 en la posición 0 y avanza a la posición 1 con estado $1D'$:

$$\begin{pmatrix} 1 & c_1 & \cdots & c_k \\ 1D' \end{pmatrix}.$$

Por las transiciones de $1D'$, la cabeza retrocede a la posición 0 en ambos casos, escribiendo 1 en la posición 1 si $c_1 = 0$, o conservándolo sin cambios si $c_1 = 1$. En cualquier caso, la posición 1 queda en 1, y se distinguen dos subcasos según el estado resultante.

- Si el estado es $1G$: la configuración es

$$\begin{pmatrix} 1 & 1 & c_2 & \cdots c_k \\ 1G \end{pmatrix},$$

que coincide exactamente con el caso $c_0 = 1, c_1 = 1$ del **Estado inicial** $1G$ (Sección 6.2.1.4), ya analizado. **Descartada.**

- **Si el estado es $3G$:** la configuración es

$$\begin{pmatrix} 1 & 1 & c_2 & \cdots & c_k \\ 3G \end{pmatrix},$$

que coincide con el **Estado inicial** $3G$ (Sección ??), ya demostrado descartado para cualquier contenido de la cinta.

Por tanto, desde el estado $1D$ con $c_0 = 0$, la máquina siempre queda descartada, independientemente del contenido del resto de la cinta.

Toda configuración con estado inicial $1D$ queda descartada. ■

6.2.2 Estados descartados por ser más lentos que $Q(k)$

6.2.2.1 Estado inicial $3D$

Por las transiciones:

$$(3D, 0) \rightarrow (1D, 0, \rightarrow), \quad (3D, 1) \rightarrow (3D', 0, \rightarrow).$$

- Si $c_0 = 1$: por la transición $(3D, 1) \rightarrow (3D', 0, \rightarrow)$, la cabeza escribe **0** en posición 0 y avanza a posición 1 con estado $3D'$. Se distinguen dos sub-casos según c_1 :
 - Si $c_1 = 0$: la cabeza retrocede a posición 0 con estado $3G$. Por la transición $(3G, 0) \rightarrow (1G, 0, \leftarrow)$, la cabeza retrocede a posición -1 . **Descartada. ■**
 - Si $c_1 = 1$: la cabeza escribe **0** en posición 1 y avanza a posición 2 con estado $1D'$. Por las transiciones de $1D'$, en ambos casos la cabeza escribe 1 en posición 2 y retrocede a posición 1. Se distinguen dos sub-sub-casos según el estado resultante:
 - * Si el estado es $1G$: por la transición $(1G, 0) \rightarrow (1G', 1, \leftarrow)$, la cabeza escribe 1 en posición 1 y retrocede a posición 0. Por la transición $(1G', 0) \rightarrow (1D, 1, \rightarrow)$, la cabeza escribe 1 en posición 0 y avanza a posición 1. Por la transición $(1D, 1) \rightarrow (3G', 1, \leftarrow)$, la cabeza retrocede a posición 0. Por la transición $(3G', 1) \rightarrow (1G', 0, \leftarrow)$, la cabeza retrocede a la posición -1 . **Descartada. ■**
 - * Si el estado es $3G$: por la transición $(3G, 0) \rightarrow (1G, 0, \leftarrow)$, la cabeza retrocede a posición 0. Por la transición $(1G, 0) \rightarrow (1G', 1, \leftarrow)$, la cabeza escribe 1 en posición 0 y retrocede a la posición -1 . **Descartada. ■**

- Si $c_0 = 0$: por la transición $(3D, 0) \rightarrow (1D, 0, \rightarrow)$, la cabeza escribe 0 (no cambia) y avanza a posición 1 con estado $1D$. Se distinguen dos sub-casos según c_1 :
 - Si $c_1 = 1$: por la transición $(1D, 1) \rightarrow (3G', 1, \leftarrow)$, la cabeza escribe 1 y retrocede a posición 0. Por la transición $(3G', 0) \rightarrow (3D, 0, \rightarrow)$, la cabeza avanza a posición 1, donde ahora hay un 1. Esto reproduce exactamente la Rama A.1 del estado $3G'$, ya demostrada como descartada. Descartada. ■
 - Si $c_1 = 0$: por la transición $(1D, 0) \rightarrow (1D', 1, \rightarrow)$, la cabeza escribe 1 en posición 1 y avanza a posición 2 con estado $1D'$. La configuración queda:

$$\begin{pmatrix} 0 & 1 & c_2 & \cdots & c_k \\ & & 1D' & & \end{pmatrix}.$$

Esto corresponde al patrón $D'_1(n)$ del Lema 6.1 con $n = k - 3$ y $s = c_k$, cuya ejecución tarda $T(k - 3)$ pasos adicionales. El tiempo total desde el estado inicial $3D$ es:

$$2 + T(k - 3).$$

Como T es estrictamente creciente y $k - 3 > k - 4$ para todo $k \geq 4$, se tiene $T(k - 3) > T(k - 4)$, y por tanto:

$$2 + T(k - 3) > 3 + T(k - 4) = Q(k).$$

Esta es la configuración más rápida alcanzable desde el estado inicial $3D$, pero aun así supera a $Q(k)$. Más lenta que $Q(k)$. ■

Toda configuración con estado inicial $3D$ queda descartada o tarda más que $Q(k)$. ■

6.2.3 Estado inicial $3G'$: configuración óptima

Del análisis de ramas del estado $3G'$:

- $c_0 = 1$: $(3G', 1) \rightarrow (1G', 0, \leftarrow)$, la cabeza va a posición -1 en el primer paso. Descartada. ■
- $c_0 = 0, c_1 = 1, c_2 = 0$ (**Rama B.0**): tras 5 pasos la cabeza alcanza la posición -1 . Descartada. ■
- $c_0 = 0, c_1 = 1, c_2 = 1$ (**Rama B.1**): la cabeza retrocede desde posición 3 con estado $1D'$ y alcanza la posición -1 en 9 pasos (sufijo 1) o 13 pasos (sufijo 0), para todo $k \geq 4$. Descartada. ■
- $c_0 = 0, c_1 = 0, c_2 = 1$ (**Rama A.1**): tras 3 pasos la cinta tiene 1 en posición 2 y la dinámica reproduce la Rama B, ya descartada. La cabeza alcanza -1 en 11 pasos (sufijo 0) o 17 pasos (sufijo 1), para todo $k \geq 5$. Descartada. ■

- $c_0 = 0, c_1 = 0, c_2 = 0$ (**Rama A.0**): los tres primeros pasos producen:

$$(3G', 0) \rightarrow (3D, 0, \rightarrow), \quad (3D, 0) \rightarrow (1D, 0, \rightarrow), \quad (1D, 0) \rightarrow (1D', 1, \rightarrow).$$

La cinta queda con **1** en posición 2, cabeza en posición 3, estado $1D'$:

$$\begin{pmatrix} 0 & 0 & \mathbf{1} & 0 & \cdots \\ & & & 1D' & \end{pmatrix}$$

Esto corresponde exactamente al patrón $D'_1(k-4)$ del Lema 6.1. La cabeza avanza desde posición 3 hasta posición k en exactamente $T(k-4)$ pasos. El tiempo total es:

$$Q(k) = 3 + T(k-4). \quad \checkmark$$

6.2.3.1 Estado inicial $1G'$

Por las transiciones:

$$(1G', 0) \rightarrow (1D, 1, \rightarrow), \quad (1G', 1) \rightarrow (3D, 1, \rightarrow).$$

Caso $c_0 = 0$.

La cabeza escribe **1** en posición 0, avanza a posición 1 con estado $1D$. Esto reproduce exactamente la situación del estado inicial $1D$ con $c_0 = 0$, ya analizado, cuyo tiempo mínimo es $1 + T(k-2) > Q(k)$. Descartada. ■

Caso $c_0 = 1$.

La cabeza escribe 1 (no cambia), avanza a posición 1 con estado $3D$. Leemos c_1 . Se abren dos sub-ramas:

- $c_1 = 1$: $(3D, 1) \rightarrow (3D', 0, \rightarrow)$, avanza a posición 2 con estado $3D'$, que siempre alcanza -1 . Descartada. ■
- $c_1 = 0$: $(3D, 0) \rightarrow (1D, 0, \rightarrow)$, avanza a posición 2 con estado $1D$. Luego, si $c_2 = 0$: $(1D, 0) \rightarrow (1D', 1, \rightarrow)$, escribiendo **1** en posición 2, avanzando a posición 3 con estado $1D'$. Esto corresponde exactamente al patrón $D'_1(k-4)$ del Lema 6.1, cuya ejecución tarda $T(k-4)$ pasos adicionales. El tiempo total es:

$$3 + T(k-4) = Q(k). \quad \checkmark$$

Si $c_2 = 1$: $(1D, 1) \rightarrow (3G', 1, \leftarrow)$, retrocede a posición 1 y entra en la dinámica de Rama A.1 del estado $3G'$, ya descartada. Descartada. ■

Por lo tanto, las únicas configuraciones con estado inicial $1G'$ que alcanzan la posición k en tiempo $Q(k)$ son aquellas con $c_0 = 1, c_1 = 0, c_2 = 0$, es decir la familia

$$\begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 1G' \end{pmatrix}.$$

Todas las demás configuraciones con estado inicial $1G'$ se descartan o tardan más que $Q(k)$.

Antes de abordar el caso general de trayectorias unbounded, notamos que el análisis anterior, realizado íntegramente para el lado derecho, se traslada de manera inmediata al lado izquierdo. En efecto, BINSMART exhibe una simetría espacial entre sus estados de avance y retroceso: intercambiar sistemáticamente los estados $1D \leftrightarrow 1G$ y $3D \leftrightarrow 3G$ en cada transición de la máquina, junto con la reflexión de la cinta respecto a la posición inicial, transforma toda trayectoria que descubre k celdas hacia la derecha en una trayectoria que descubre k celdas hacia la izquierda, preservando exactamente el número de pasos empleado. Por tanto, si una configuración inicial alcanza la posición k en tiempo $Q(k)$ sin visitar la posición -1 , su configuración simétrica alcanza la posición $-k$ en el mismo tiempo $Q(k)$ sin visitar la posición 1 , y viceversa. En particular, esto implica que $Q(k)$ es también el tiempo mínimo para descubrir k celdas exclusivamente hacia la izquierda, y que las familias óptimas identificadas para el lado derecho tienen sus contrapartes simétricas exactas del lado izquierdo. Con esta equivalencia establecida, resta únicamente considerar el caso en que ambas direcciones se exploran simultáneamente.

6.2.4 Configuraciones que descubren celdas en ambas direcciones

Hasta ahora hemos demostrado que $Q(k)$ es el tiempo mínimo para alcanzar la posición k entre todas las configuraciones que no visitan la posición -1 . En esta subsección extendemos el resultado al caso más general: ninguna configuración inicial, incluyendo aquellas cuya trayectoria descubre celdas en *ambas* direcciones, puede descubrir k celdas nuevas en menos de $Q(k)$ pasos.

Definición 6.4. Sea x la posición inicial de la cabeza. Decimos que una trayectoria es *unbounded* si en algún instante la cabeza visita una posición estrictamente a la derecha de x y en otro instante visita una posición estrictamente a la izquierda de x . Si la posición más lejana visitada a la derecha es $x + c$ y la más lejana a la izquierda es $x - j$, decimos que la trayectoria descubre c celdas a la derecha y j celdas a la izquierda, con $c + j = k$.

Antes de enunciar el resultado principal, establecemos el siguiente lema auxiliar, que cuantifica el costo mínimo de explorar un segmento desde uno de sus extremos.

Lema 6.5 (Costo de segmento). *Sea $\ell \geq 1$. Para cualquier configuración en la que la cabeza se encuentra en un extremo de un segmento de ℓ celdas no visitadas y debe alcanzar el otro extremo del mismo*

segmento, el tiempo necesario es al menos $Q(\ell)$.

Demostración. Por simetría espacial, el problema de recorrer ℓ celdas hacia la izquierda desde el extremo derecho es equivalente al de recorrer ℓ celdas hacia la derecha desde el extremo izquierdo. Este último es exactamente el problema analizado en la sección anterior, cuyo tiempo mínimo es $Q(\ell) = 3 + T(\ell - 4)$, demostrado por el análisis de ramas. $\square$

Lema 6.6 (Costo de retorno). *Sea $j \geq 1$. Si una trayectoria parte en x , alcanza primero la posición $x + c$ y luego debe descubrir j celdas estrictamente a la izquierda de x , entonces el tiempo total es al menos*

$$Q(c) + Q(c + j).$$

Demostración. Procedemos por inducción sobre j .

Base $j = 1$.

Sea t_c el primer instante en que la cabeza alcanza $x + c$. Por el análisis de ramas de la sección anterior (Caso 1), llegar a $x + c$ sin visitar el lado izquierdo requiere al menos $Q(c)$ pasos, luego

$$t_c \geq Q(c).$$

A partir del instante t_c , la cabeza se encuentra en $x + c$ y aún no ha descubierto ninguna celda a la izquierda de x . Para descubrir la celda $x - 1$, la cabeza debe recorrer el segmento $[x - 1, x + c]$, que tiene longitud $c + 1$, partiendo desde su extremo derecho. Por el Lema 6.5, ese recorrido requiere al menos $Q(c + 1)$ pasos adicionales. Por tanto

$$t^* \geq t_c + Q(c + 1) \geq Q(c) + Q(c + 1),$$

que es la cota requerida para $j = 1$.

Paso inductivo.

Supongamos que el enunciado es cierto para algún $j \geq 1$, es decir, que toda trayectoria que parte en x , alcanza $x + c$ y luego descubre j celdas a la izquierda, tarda al menos $Q(c) + Q(c + j)$.

Consideremos ahora una trayectoria que, tras las mismas condiciones, debe descubrir $j + 1$ celdas a la izquierda. Sea t_j el primer instante en que la celda $x - j$ es descubierta. Por hipótesis inductiva aplicada a la misma trayectoria hasta ese instante,

$$t_j \geq Q(c) + Q(c + j).$$

Para descubrir la celda adicional $x - (j + 1)$, la cabeza se encuentra en alguna posición del segmento

$[x - j, x + c]$ en el instante t_j . Debe extenderse una posición más hacia la izquierda, cubriendo el segmento $[x - (j + 1), x + c]$ de longitud $c + j + 1$ desde el extremo derecho. Por el Lema 6.5, ese recorrido adicional requiere al menos $Q(c + j + 1)$ pasos.

Sin embargo, observamos que desde el instante t_c la cabeza debe aún alcanzar $x - (j + 1)$. El tiempo restante satisface

$$t^* - t_c \geq Q(c + j + 1),$$

por aplicación del Lema 6.5 al segmento completo $[x - (j + 1), x + c]$ de longitud $c + j + 1$ partiendo desde $x + c$. Sumando con $t_c \geq Q(c)$, se obtiene

$$t^* \geq Q(c) + Q(c + j + 1),$$

que es la cota requerida para $j + 1$. □

Teorema 6.7 (Optimalidad global de $Q(k)$). *Para todo $k \geq 1$, toda configuración inicial que descubre k celdas nuevas tarda al menos $Q(k)$ pasos. En particular, ninguna trayectoria zigzag puede igualar el tiempo de la configuración óptima.*

Demostración. Sea $\mathcal{C}$ una configuración inicial arbitraria, sea x la posición inicial de la cabeza, y sea t^* el primer instante en que la trayectoria ha descubierto exactamente k posiciones nuevas. Distinguiamos dos casos.

Caso 1: la trayectoria no es unbounded.

La cabeza permanece en un solo lado de x y debe alcanzar la posición $x + k$ (o $x - k$ por simetría). Por el análisis de ramas de la sección anterior, el tiempo mínimo es exactamente

$$Q(k) = 3 + T(k - 4),$$

alcanzado únicamente por las familias óptimas

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 3G' & & & & \end{pmatrix} \quad \text{o} \quad \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1G' & & & & \end{pmatrix}.$$

Por tanto $t^* \geq Q(k)$.

Caso 2: la trayectoria es unbounded.

La trayectoria descubre $c \geq 1$ celdas a la derecha y $j \geq 1$ celdas a la izquierda, con $c + j = k$. Sin pérdida

de generalidad, supongamos que la trayectoria alcanza primero el lado derecho; el otro orden es análogo por simetría espacial.

Por el Lema 6.6 aplicado con estos valores de c y j , el tiempo total satisface

$$t^* \geq Q(c) + Q(c + j) = Q(c) + Q(k).$$

Como $c \geq 1$, se tiene $Q(c) \geq Q(1) = 1$ por la Observación 6.2 (valor verificado directamente, sin recurrir a la fórmula $3 + T(k - 4)$, que solo aplica para $k \geq 4$), y por tanto $t^* \geq Q(k) + 1 > Q(k)$.

Ninguna trayectoria unbounded puede alcanzar el tiempo óptimo $Q(k)$.

En ambos casos $t^* \geq Q(k)$, con igualdad solo en el Caso 1 para las familias óptimas. Por tanto, $Q(k)$ es el tiempo mínimo global para descubrir k celdas nuevas en binSMART, independientemente de la dirección en que se exploren. $\square$

Capítulo 7

Discusión

7.1 Sobre la construcción de la toolbox

El desarrollo de la toolbox de esta tesis se basa en dos antecedentes complementarios. Por una parte, el Lemma 1 de BINSMART [4] proporciona la descripción cualitativa de los ocho movimientos elementales de la máquina. Dicho lema demuestra, mediante inducción sobre n , que los movimientos $D_1(n)$, $G_1(n)$, $D'_1(n)$, $G'_1(n)$, $D_3(n)$, $G_3(n)$, $D'_3(n)$ y $G'_3(n)$ ocurren efectivamente, pero no cuantifica el número de pasos necesarios para realizar cada transición.

Por otra parte, el trabajo sobre la dinámica de SMART presentado en [14] constituye el antecedente metodológico directo de esta investigación. En dicho trabajo, los movimientos cualitativos de SMART se reformulan mediante funciones que expresan su costo temporal exacto. Esta reformulación permite utilizar la estructura recursiva de los movimientos para obtener fórmulas cerradas mediante inducción y, posteriormente, aplicar dichas fórmulas al estudio de propiedades dinámicas más finas, como la sincronización de configuraciones en el contexto de weak mixing.

La toolbox desarrollada aquí adapta ese procedimiento al caso de BINSMART. En particular, se toma la descomposición recursiva de los movimientos establecida en [4] y se incorpora a cada submovimiento su costo temporal correspondiente. De esta manera, las transiciones de escala n se expresan como composiciones de movimientos de escalas menores y de pasos elementales, cuya duración puede sumarse explícitamente. El resultado es la obtención de las funciones cerradas $T(n)$, $M(n)$ y $Z(n)$, que cuantifican el costo temporal de los movimientos utilizados en el análisis de BINSMART.

Por tanto, el aporte de esta tesis no consiste únicamente en repetir la descripción cualitativa de los movimientos de [4], sino en extenderla mediante una cuantificación temporal explícita. El procedimiento utilizado sigue el enfoque metodológico de [14], pero se aplica a la máquina binaria y a los movimientos específicos de BINSMART. Esta adaptación permite utilizar posteriormente la toolbox para determinar el tiempo mínimo necesario para descubrir nuevas celdas de la cinta y para comparar trayectorias unidireccionales con trayectorias que cambian de dirección.

7.2 Sobre la velocidad logarítmica de binSMART

Resolviendo la recurrencia que define la toolbox, se obtiene la fórmula cerrada

$$Q(k) = 3 + T(k-4) = \frac{1}{6} \left(-(-1)^{k-3} + 2^{k+1} + 3 \right), \quad T(n) = \frac{1}{6} \left(-(-1)^{n+1} + 2^{n+5} - 15 \right).$$

El término dominante de $Q(k)$ es 2^{k+1} ; por tanto,

$$Q(k) = \Theta(2^k).$$

En consecuencia, el tiempo necesario para descubrir k celdas nuevas crece exponencialmente en función de k . Al invertir esta relación, se obtiene que el número máximo de celdas nuevas descubiertas en tiempo t satisface

$$m(t) = \log_2(t) + O(1).$$

Así, la exploración de BINSMART presenta crecimiento logarítmico en base 2 y, en particular, no tiene velocidad positiva.

Este resultado permite situar a BINSMART dentro de la tricotomía propuesta en [7], que distingue entre máquinas con movimiento acotado, máquinas aperiódicas distorsionadas y máquinas con velocidad positiva. En el caso distorsionado, el crecimiento del soporte satisface las cotas

$$m(t) = \Omega(\log t) \quad \text{y} \quad m(t) = O\left(\frac{t}{\log t}\right).$$

Puesto que

$$m(t) = \Theta(\log t),$$

BINSMART se sitúa en el extremo inferior de dicho régimen: alcanza el orden de crecimiento logarítmico permitido por la cota inferior de la tricotomía.

El interés de este resultado no consiste en llenar un vacío respecto de la existencia de máquinas con tal comportamiento, pues SMART ya proporciona un ejemplo de máquina aperiódica y distorsionada dentro de este marco. El aporte específico para BINSMART consiste en obtener una caracterización cuantitativa explícita de su crecimiento mediante la función cerrada $Q(k)$ y en demostrar que dicha función corresponde al tiempo mínimo global para descubrir k celdas nuevas. De este modo, BINSMART aporta otro ejemplo concreto dentro del régimen distorsionado, ahora sobre un alfabeto binario, cuya tasa de exploración queda determinada con precisión aritmética a partir de la toolbox desarrollada en esta tesis.

7.3 Sobre el camino abierto hacia weak mixing

El resultado de $Q(k)$ y su optimalidad frente a trayectorias unbounded no demuestran que BINSMART sea weak mixing, pero sí constituyen el fundamento cuantitativo necesario para abordar esa pregunta: la técnica de *embedding* utilizada por Torres-Avilés para demostrar weak mixing en SMART requiere sincronizar dos configuraciones locales distintas dentro de una ventana de tiempo común [14], y dicha sincronización exige conocer exactamente cuántos pasos toma cada movimiento involucrado —precisamente la información que esta tesis aporta para BINSMART y que el Lemma 1 original de [4] no proveía.

Cabe señalar que existe una vía alternativa, de naturaleza puramente algebraica, para responder la misma pregunta: dado que el t-shift de BINSMART es la clausura del punto fijo de una sustitución primitiva σ (Teorema 3 de [4]), es posible estudiar weak mixing mediante el análisis espectral de la matriz de sustitución asociada a σ , examinando si sus valores propios distintos del valor de Perron-Frobenius son o no raíces de la unidad. Esta vía no requiere ningún análisis cuantitativo de tiempos de recorrido y es, en principio, independiente del enfoque desarrollado en esta tesis. Ambos caminos —el cuantitativo, vía la toolbox aquí construida, y el algebraico, vía la matriz de sustitución— quedan abiertos como líneas de trabajo futuro, y no es evidente a priori cuál resultaría más directo para BINSMART.

Más allá de weak mixing, la toolbox abre también la puerta a estudiar la mezcla topológica (*topological mixing*) de BINSMART, una propiedad estrictamente más fuerte que weak mixing dentro de la jerarquía revisada en el Capítulo 2 [14]. Es importante notar que ni siquiera para SMART se ha determinado si la máquina es topológicamente mixing —el resultado de [14] establece weak mixing, pero no mixing topológico—, de modo que abordar esta propiedad en BINSMART no solo respondería una pregunta abierta sobre esta máquina en particular, sino que se adelantaría a una pregunta que ni siquiera ha sido resuelta para su máquina de origen.

Capítulo 8

Conclusiones

8.1 Síntesis de resultados

Esta tesis desarrolló una toolbox analítica que cuantifica, mediante fórmulas cerradas demostradas por inducción, el costo temporal exacto de los ocho movimientos elementales que el Lemma 1 [4] solo certificaba de forma cualitativa. Los resultados centrales obtenidos son:

- **Toolbox cuantitativa.** Las funciones cerradas $T(n)$, $M(n)$ y $Z(n)$ (Lemas 6.1–6.3) cuantifican el costo exacto de cada uno de los ocho movimientos elementales de binSMART, cerrando la laguna dejada por el Lemma 1 puramente existencial [4].
- **Tiempo mínimo global $Q(k)$.** A partir de la toolbox se determinó el tiempo mínimo $Q(k) = 3 + T(k - 4)$ necesario para que binSMART descubra k celdas nuevas de la cinta, identificando explícitamente las familias de configuraciones óptimas que lo alcanzan (Sección 6.2), y estableciendo mediante los lemas de costo de segmento y costo de retorno (Lemas 6.5 y 6.6) que ninguna trayectoria unbounded puede igualar dicho tiempo (Teorema 6.7).
- **Velocidad logarítmica.** Puesto que $Q(k) = \Theta(2^k)$, se obtiene $m(t) = \Theta(\log_2 t)$: binSMART constituye un ejemplo explícito y cuantificado —no solo un argumento de existencia— de máquina mínima aperiódica que satura la cota inferior del régimen distorsionado en la tricotomía de Guillon y Salo [7], respondiendo así a la pregunta que dichos autores dejan explícitamente abierta sobre qué tasas de crecimiento intermedio son efectivamente realizables.

Estos resultados confirman la hipótesis planteada en el Capítulo 4: existe una familia de configuraciones óptima, expresable mediante una fórmula cerrada dependiente de k , y esta optimalidad se sostiene a nivel global y no solo dentro de una clase restringida de trayectorias.

8.2 Alcance y limitaciones

Como se estableció en el Capítulo 3, esta tesis se restringió deliberadamente a construir el fundamento cuantitativo —la toolbox y la función $Q(k)$ — y no abordó la demostración ni refutación de que binSMART sea weak mixing. Esta limitación no es un vacío accidental, sino una decisión de alcance: la técnica de embedding empleada por [14] para demostrar weak mixing en SMART exige sincronizar dos configuraciones locales distintas dentro de una ventana de tiempo común, y dicha sincronización requiere precisamente el tipo de cuantificación temporal exacta que esta tesis aporta para binSMART. La toolbox construida es, por tanto, condición necesaria —aunque no suficiente— para abordar weak mixing en binSMART.

8.3 Recomendaciones para trabajo futuro

A partir de los resultados obtenidos, se identifican dos líneas de investigación concretas y complementarias:

1. **Vía cuantitativa (embedding).** Emplear la toolbox construida en esta tesis para replicar sobre binSMART la técnica de embedding aplicada por [14] sobre SMART, sincronizando pares de configuraciones locales dentro de una ventana de tiempo común mediante las fórmulas $T(n)$, $M(n)$, $Z(n)$ y $Q(k)$ aquí demostradas, con el objetivo de determinar si binSMART es topológicamente débilmente mezclante en su modelo TMT.
2. **Vía algebraica (espectral).** Dado que el t -shift de binSMART es la clausura del punto fijo de una sustitución primitiva σ (Teorema 3 [4], [Coversación personal]), explorar si sus valores propios distintos del valor propio de Perron-Frobenius son o no raíces de la unidad —una vía independiente que no requiere análisis cuantitativo de tiempos de recorrido, y que podría resultar más directa que el enfoque combinatorio de esta tesis.
3. **Mixing topológico.** Más allá de weak mixing, se recomienda explorar la propiedad de mezcla topológica (*topological mixing*) en binSMART, estrictamente más fuerte dentro de la jerarquía revisada en el Capítulo 2. Cabe destacar que ni siquiera para SMART se ha determinado si la máquina es topológicamente mixing —el resultado de [14] establece únicamente weak mixing—, de modo que resolver esta pregunta para binSMART se adelantaría a una cuestión abierta incluso para su máquina de origen.
4. **Generalización del marco analítico.** Explorar si la metodología de esta tesis —convertir un Lemma de existencia cualitativo en una toolbox de costos temporales cerrados, demostrados por inducción— es extensible a otras máquinas de Turing binarias aperiódicas y reversibles, más allá del

caso particular de la velocidad de exploración aquí abordado.

Bibliografía

- [1] Ethan Akin. *The General Topology of Dynamical Systems*. Vol. 1. Graduate Studies in Mathematics. American Mathematical Society, 1993.
- [2] Vincent D. Blondel, Julien Cassaigne y Codrin Nichitiu. “On the presence of periodic configurations in Turing machines and in counter machines”. En: *Theoretical Computer Science* 289.1 (2002), págs. 573-590.
- [3] Julien Cassaigne, Nicolas Ollinger y Rodrigo Torres-Avilés. “A small minimal aperiodic reversible Turing machine”. En: *Journal of Computer and System Sciences* 84 (2017), págs. 288-301.
- [4] Pablo Concha-Vega y Rodrigo Torres-Avilés. “A binary complete and aperiodic Turing machine”. En: *International Journal of Unconventional Computing* 16.1 (2020), págs. 19-39.
- [5] Anahí Gajardo y Pierre Guillon. “Zigzags in Turing machines”. En: *Computer Science – Theory and Applications (CSR 2010)*. Vol. 6072. Lecture Notes in Computer Science. Kazan, Russia: Springer, 2010, págs. 109-119.
- [6] Anahí Gajardo, Nicolas Ollinger y Rodrigo Torres-Avilés. “Some undecidable problems about the trace-subshift associated to a Turing machine”. En: *Discrete Mathematics & Theoretical Computer Science* 17. Automata, Logic and Semantics (2015).
- [7] Pierre Guillon y Ville Salo. “Distortion in one-head machines and cellular automata”. En: *Cellular Automata and Discrete Complex Systems (AUTOMATA 2017)*. Lecture Notes in Computer Science. Springer, 2017, págs. 120-138.
- [8] Philip K. Hooper. “The Undecidability of the Turing Machine Immortality Problem”. En: *The Journal of Symbolic Logic* 31.2 (1966), págs. 219-234.
- [9] Emmanuel Jeandel. “Computability of the Entropy of One-Tape Turing Machines”. En: *31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014)*. Vol. 25. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2014, págs. 421-432.
- [10] Jarkko Kari y Nicolas Ollinger. “Periodicity and immortality in reversible computing”. En: *International Symposium on Mathematical Foundations of Computer Science*. Springer, 2008, págs. 419-430.

- [11] Petr Kůrka. “On topological dynamics of Turing machines”. En: *Theoretical Computer Science* 174.1-2 (1997), págs. 203-216.
- [12] Ville Lukkarila. “On Undecidable Dynamical Properties of Reversible One-Dimensional Cellular Automata”. Tesis doct. University of Turku, 2010.
- [13] Ville Salo e Ilkka Törmä. *A Physically Universal Turing Machine*. arXiv preprint. 2020. arXiv: 2004.02037.
- [14] Rodrigo Torres-Avilés. “Topological mixing notions on Turing machine dynamical systems”. En: *Information and Computation* 285 (2022), pág. 104915.
- [15] Rodrigo Torres-Avilés. “Undecidability of the speed positiveness problem in reversible and complete Turing machines”. En: *Journal of Computer and System Sciences* 121 (2021), págs. 49-56.
- [16] Alan M. Turing. “On Computable Numbers, with an Application to the Entscheidungsproblem”. En: *Proceedings of the London Mathematical Society* 2.1 (1936), págs. 230-265.