



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

**“UNA DEMOSTRACIÓN COMBINATORIA DE
IMPOSIBILIDAD DEL RENOMBRADO”**

T E S I S

QUE PARA OBTENER EL GRADO DE:

**MAESTRO EN CIENCIAS
(COMPUTACIÓN)**

P R E S E N T A:

ARMANDO CASTAÑEDA ROJANO

DIRECTOR DE TESIS

DR. SERGIO RAJSBAUM GORODEZKY

México, D.F.

2007.



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

Quiero agradecer a Sergio Rajsbaum por permitirme trabajar a su lado, por dedicarme tiempo y tenerme paciencia en el desarrollo de la tesis.

A Javier Bracho, Jorge Urrutia, Carlos Velarde y Luis Morales, por sus observaciones y comentarios que enriquecieron este trabajo.

A mi familia y amigos, por ser tan especiales y por estar siempre conmigo cuando los he necesitado.

A Norma, por apoyarme, quererme y hacerme tan feliz.

Al CONACYT, por apoyarme económicamente durante mis estudios de maestría.

Índice general

1. Introducción	1
1.1. Motivación y Problemática	1
1.1.1. Los Sistemas Distribuidos	1
1.1.2. El Renombrado y la Relación con la Topología	3
1.2. Objetivos y Contribución	6
1.3. Organización de la Tesis	7
2. El problema del Renombrado	9
2.1. Modelos de Cómputo	9
2.1.1. Memoria Compartida	9
2.1.2. Envío de Mensajes	12
2.1.3. Tareas	13
2.1.4. Ejecuciones <i>Immediate Snapshot</i>	13
2.2. El Renombrado	14
2.2.1. Especificación	15
2.2.2. Cotas	16
2.2.3. Dos Algoritmos de Renombrado	17
3. Los Sistemas Distribuidos y la Topología	21
3.1. Conceptos Topológicos	21
3.1.1. Conceptos Básicos	21
3.1.2. Pseudovariedades	24
3.1.3. El Lema del Índice	25
3.1.4. Imágenes Divididas	29
3.2. Modelado de Tareas y Algoritmos	40
3.2.1. Modelado de Tareas	40
3.2.2. Modelado de Algoritmos	41
3.2.3. Condición de Solubilidad	42
3.2.4. El Complejo de las Ejecuciones <i>Immediate Snapshot</i>	42
3.2.5. El Renombrado y las Imágenes Divididas	44
4. Caracterización del Número de Simplejos Monocromáticos	47
4.1. Estrategia de la Demostración	47
4.2. Contando Simplejos Monocromáticos	49
4.3. Caracterización Combinatoria	52
4.4. Calculando el Valor k	61
4.5. Cuando el Índice es Igual a Cero	66
4.5.1. Estructura de \mathbb{K}^5	66

4.5.2. Índice Igual a Cero	67
5. Consecuencias y Conclusiones	73
5.1. Consecuencias	73
5.1.1. Demostración Combinatoria de Imposibilidad del Renombrado	73
5.1.2. Implicaciones de \mathbb{K}^5	75
5.2. Conclusiones	76
5.3. Trabajo Futuro	76

Índice de figuras

1.1. Control de tráfico aéreo.	2
1.2. Representación gráfica de simplejos de dimensión 0,1, 2 y 3.	4
1.3. El renombrado con tres nombres.	5
1.4. El renombrado con cuatro nombres.	5
2.1. Memoria compartida.	11
2.2. Consenso binario.	13
2.3. Ejecuciones IS indistinguibles para p_1 y p_2	14
3.1. Complejos de dimensión 1 y 2.	22
3.2. Un simplejo orientado.	24
3.3. Una pseudovariedad orientada coherentemente.	25
3.4. El Lema del Índice en acción.	26
3.5. Ejemplo de una imagen dividida.	30
3.6. Un par de imágenes divididas con aristas de cruce.	31
3.7. El Lema de Sperner.	33
3.8. Un par de caminos sobre una pseudovariedad cromática.	34
3.9. Las 2-esquinas de una imagen dividida de un 2-simplejo	37
3.10. Complejos del consenso binario.	41
3.11. Complejo de ejecuciones IS en el que los procesadores están ac- tivos una vez.	43
3.12. Complejo del algoritmo <i>AlgRenombrado1</i>	44
4.1. Un ejemplo del proceso inductivo.	48
4.2. Una pseudovariedad con las tres coloraciones.	51
4.3. Complejo estrella de un vértice en la frontera.	54
4.4. Un ejemplo de la coloración inducida por c^n	63
4.5. Dos 1-pasos sobre \mathbb{K}^5	68
4.6. Un 2-paso sobre \mathbb{K}^5	69
4.7. Dos caminos entre simplejos monocromáticos de \mathbb{K}^5	70

Capítulo 1

Introducción

1.1. Motivación y Problemática

Comenzamos presentando un poco de teoría de los sistemas distribuidos, así como un ejemplo sencillo del tipo de problemas que enfrentan. Después explicaremos brevemente la fuerte relación que tienen con la topología.

1.1.1. Los Sistemas Distribuidos

Los importantes avances en las tecnologías de comunicación han provocado que los sistemas distribuidos cobren gran importancia en la actualidad. Todos los días hacemos uso de ellos y dependemos de su fiabilidad. Consideramos a un sistema distribuido como un sistema que consta de un conjunto de agentes de cómputo autónomos que afectan su comportamiento entre sí por medio del intercambio de información. Dichos agentes son comúnmente conocidos como *procesos* o *procesadores*, y cuentan con capacidad de cómputo y memoria individual. Estos *procesadores* se comunican a través del *envío de mensajes* o de una *memoria compartida*. Los sistemas que utilizan *mensajes* cuentan con un medio de comunicación por el cual los procesadores pueden enviar paquetes de información a cualquier otro procesador en el sistema, incluido él mismo. En los sistemas con memoria compartida cada procesador es dueño de un *registro*, el cual puede ser escrito únicamente por su propietario pero puede ser leído por todos los procesadores del sistema.

Una característica importante de cualquiera de los dos modelos es que pueden ser síncronos o asíncronos. En un modelo síncrono los procesadores trabajan a la misma velocidad y los retardos en la comunicación son conocidos y, por tanto, estos ejecutan sus operaciones al mismo tiempo. Es decir, todo procesador conoce de antemano el momento en el que recibirá alguna información de los demás. En cambio, en un modelo asíncrono no existe ninguna restricción en cuanto a las velocidades de los procesadores o los retardos en la comunicación. La diferencia de fondo entre estas dos concepciones es que en el modelo síncrono un procesador puede saber con exactitud cual es el estado de los demás procesadores en el sistema, mientras que en el modelo asíncrono no. Nuestro interés

se centra en los modelos asíncronos, en particular el de memoria compartida.

La relación fundamental que guardan los modelos asíncronos de envío de mensajes y memoria compartida es que son equivalentes [4] (considerando que en el de envío de mensajes únicamente pueden fallar menos de la mitad de los procesadores). A saber, si existe un algoritmo para un problema dado en cualquiera de los dos modelos, entonces también existe un algoritmo que soluciona el mismo problema en el otro modelo. Sin embargo, por su sencillez en cuanto a la estructura, frecuentemente es preferible utilizar el modelo de memoria compartida, ya que facilita el diseño de algoritmos o la obtención de resultados de imposibilidad, es decir, probar que no existe algoritmo que solucione alguna tarea.

Supongamos que queremos desarrollar un sistema autónomo de control de tráfico aéreo en el que $n + 1$ aviones tienen que decidir la altitud a la que deben volar para evitar colisiones (ver figura 1.1). ¿Cuántas altitudes distintas se necesitan para solucionar el problema? La primera respuesta que nos viene a la mente es $n + 1$. En efecto, existen condiciones para las cuales este problema se resuelve muy fácilmente. Por ejemplo, supongamos que los aviones tienen asociados los identificadores $a_0, a_1 \dots a_n$. Además, denotamos las distintas altitudes con $0, 1 \dots k$ (por supuesto que k debe ser mayor o igual a n). Si el avión a_i elige volar a altitud i , entonces el problema se soluciona haciendo uso del menor número de altitudes posibles, una por cada avión, sin la necesidad de intercambiar información. Ahora supongamos que la única forma en que se pueden distinguir los aviones entre sí es por medio del número de vuelo que están realizando, y que la única operación que tienen disponible con la información que intercambian son comparaciones lexicográficas. Es decir, un avión sólo puede saber si el nombre de algún otro es menor o mayor al suyo (los aviones ni siquiera pueden saber cual es su número de vuelo). El siguiente es un ejemplo de un algoritmo que soluciona el problema para dos aviones con números de vuelo A y B , respectivamente. En él nos referiremos a los aviones por su número de vuelo. Básicamente tenemos tres escenarios posibles:

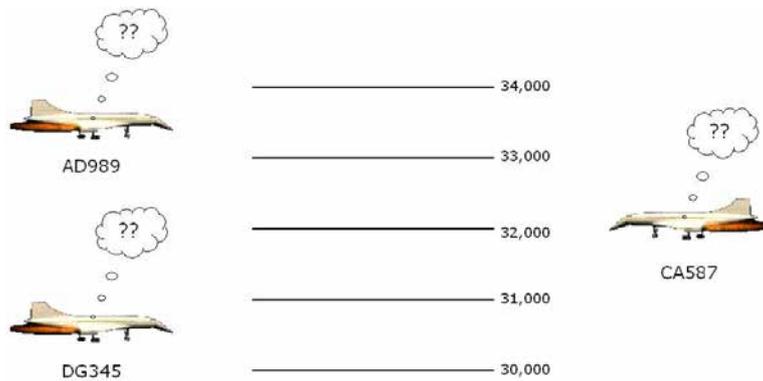


Figura 1.1: Control de tráfico aéreo.

- A no se entera del número de vuelo de B (ya sea porque tiene mayor velocidad de cómputo o por un retardo grande en la comunicación) pero B si se entera del número de vuelo de A . Es decir, A tiene que decidir conociendo únicamente su número de vuelo y B decidirá conociendo su número de vuelo y el de A . Supongamos que un avión decide 0 cuando no tiene contra que comparar su número de vuelo y que decide 1 cuando su número de vuelo es mayor que el del otro avión. Por lo tanto, A y B deciden volar a las altitudes 0 y 1, respectivamente.
- B no se entera del número de vuelo de A pero A si se entera del número de vuelo de B . Debido a que cada avión ni siquiera puede saber cual es el número de vuelo que está realizando, y acorde al caso anterior, B debe decidir volar a altitud 0. Sin embargo, A debe decidir a que altura volar sabiendo que su nombre es menor que el del otro avión. Supongamos que para éste caso decide 2.
- A y B se enteran del número de vuelo del otro avión. De acuerdo con los dos casos anteriores, A y B deciden 2 y 1, respectivamente.

El algoritmo anterior resuelve el problema utilizando tres altitudes distintas. La pregunta es, ¿se puede desarrollar un algoritmo que solucione el problema para dos aviones haciendo uso únicamente de dos altitudes? La respuesta es no. Para que esto sucediera se necesita que un avión decida 1 cuando su número de vuelo es menor que el del otro avión. De esta forma, en el segundo caso los aviones decidirán volar a altitudes 0 y 1. Sin embargo, en el tercer caso los dos aviones decidirán volar a altitud 1. El problema fundamental es el siguiente: el avión B no tiene certeza de cual es el escenario en el que está A , es decir, no puede estar seguro de si A sabe que B es el número de vuelo del otro avión y, por tanto, no puede estar seguro de como se comportará.

1.1.2. El Renombrado y la Relación con la Topología

El problema expuesto en la sección anterior, comúnmente conocido como el problema del *renombrado*, es un ejemplo de los problemas de coordinación a los que se enfrentan constantemente los sistemas distribuidos. La especificación es a groso modo la siguiente. En el problema del *k-renombrado* [5] cada procesador comienza con un nombre único, tomado de un dominio muy grande de nombres originales. El objetivo es que cada procesador elija, de forma irreversible, un nombre de un dominio de nombres nuevos, $[0 \dots k]$, lo más pequeño posible, de forma tal que cualesquiera dos nombres nuevos sean distintos.

El problema del *k-acuerdo* es otro ejemplo de los problemas que solucionan los sistemas distribuidos. Este es la generalización del clásico problema del *consenso*. En el *k-acuerdo* cada procesador comienza con un valor de entrada que propondrá a los demás procesadores. El objetivo es que los procesadores decidan a lo más k valores distintos de entre los valores de entrada propuestos.

Por sus fuertes implicaciones, estos dos problemas son considerados fundamentales en al área de los sistemas distribuidos. Por ejemplo, representan abstracciones de los problemas de coordinación que tienen que solventar los

sistemas de transacciones bancarias que cuentan con una base de datos distribuida. Para estos sistemas es de suma importancia asegurar que las distintas computadoras donde reside la base de datos ejecuten secuencias de operaciones idénticas, con el fin de garantizar la consistencia de la misma (una descripción más detallada de las aplicaciones de estos y otros problemas se puede encontrar en [6, 20]). Además, también tienen fuertes implicaciones teóricas, ya que los diversos trabajos de investigación sobre el renombrado y el acuerdo han generado los fundamentos teóricos del cómputo distribuido. Aún más, en recientes investigaciones se ha descubierto que estos dos problemas guardan una relación importante: con una implementación del acuerdo es posible implementar el renombrado, pero no al revés [13]. Es decir, el renombrado es estrictamente más débil que el acuerdo.

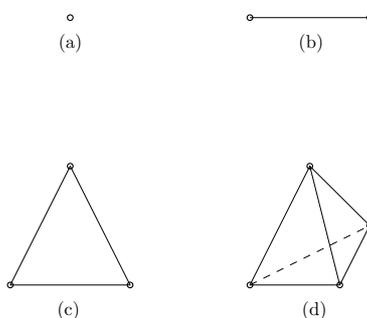


Figura 1.2: Representación gráfica de simplejos de dimensión 0,1, 2 y 3.

En el Simposio Anual sobre Teoría de la Computación (STOC) de 1993, tres grupos de investigación [18, 22, 9] independientes demostraron la fuerte relación que existe entre los sistemas distribuidos y la topología. Estos trabajos dieron como resultado una caracterización topológica de lo que es computable *libre de espera* por un sistema asíncrono de memoria compartida. Esto es, una caracterización de lo que pueden solucionar aquellos algoritmos asíncronos que se comunican a través de una memoria compartida y que son tolerantes a cualquier número de fallas de paro de los procesadores. Ellos observaron que los *estados* de los procesadores al final de una ejecución de un algoritmo pueden ser representados por un *simplejo*, el cual no es más que un conjunto finito (ver figura 1.2). Mientras que todas las ejecuciones posibles del algoritmo pueden representarse como un *complejo*, un objeto matemático construido a base de simplejos que representa la discretización de un espacio geométrico y que permite estudiar este último con mayor facilidad. El resultado fundamental es que los complejos de las ejecuciones de los algoritmos asíncronos libres de espera de memoria compartida son *sólidos*, es decir, no tienen hoyos (esta es la diferencia principal con los modelos síncronos, ya que los complejos que se generan bajo esta condición sí tienen hoyos). Esta invariante topológica es de gran importancia, ya que ha dado pie a resultados de imposibilidad del renombrado y el acuerdo. El resultado en su totalidad es realmente relevante, no solo para los sistemas distribuidos sino también para la teoría de la computación, al grado que dos de los tres artículos [18, 22] recibieron el afamado Premio Gödel, mientras que el

tercero [9] no se hizo acreedor al mismo por no haber sido publicado en revista. Inclusive, [18, 22] están considerados dentro de la lista de publicaciones más importantes en las ciencias de la computación [23].

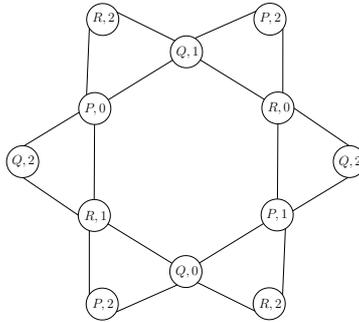


Figura 1.3: El renombrado con tres nombres.

El siguiente ejemplo proporciona una idea de la relación que guardan los sistemas distribuidos y la topología. En la Figura 1.3 se puede ver un complejo que representa las salidas de un algoritmo que soluciona el renombrado para tres procesadores, P , Q y R , con el dominio de nombres nuevos $\{0, 1, 2\}$. Observe que en el complejo están todas las posibles combinaciones en las que los tres procesadores pueden elegir los tres nombres. La Figura 1.4¹ muestra el caso para el cual se utiliza un nombre más en el dominio de nombres nuevos.

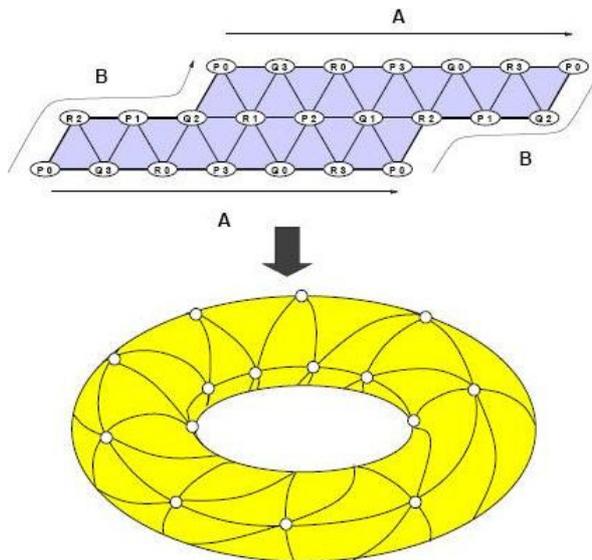


Figura 1.4: El renombrado con cuatro nombres.

¹Agradecemos a Maurice Herlihy el habernos permitido utilizar esta figura.

En [5] se presenta un algoritmo que soluciona el $2n$ -renombrado para un sistema con $n + 1$ procesadores. Esto es lo mejor que se puede obtener ya que se ha demostrado que no existe algoritmo libre de espera que solucione el $(2n - 1)$ -renombrado. La primera versión de esta demostración se presentó en el STOC de 1993 [18], mientras que la versión definitiva aparece en el Journal de la ACM en el 2000, la cual se basa, según sus autores, en la demostración y herramientas que se presentan en [19]. A diferencia del $(n - 1)$ -acuerdo, para el cual un sencillo argumento de conteo basado en el conocido *lema de Sperner* es suficiente para entender la imposibilidad del problema [9, 3], para comprender la demostración del $(2n - 1)$ -renombrado es necesario poseer conocimientos de topología algebraica. Una pregunta natural que nos podemos hacer es ¿por qué el renombrado, que es estrictamente más débil que el acuerdo, tiene una demostración de imposibilidad bastante más complicada? No obstante, se han realizado avances hacia una demostración combinatoria más simple [3], aunque la parte crucial de la misma sigue dependiendo de argumentos del área de la topología algebraica.

1.2. Objetivos y Contribución

El objetivo principal de la tesis consiste en estudiar el complejo de las ejecuciones de un algoritmo libre de espera que soluciona el renombrado en el modelo asíncrono de memoria compartida, con el fin de encontrar una demostración combinatoria más simple de imposibilidad del renombrado. Para éste estudio hacemos uso de las herramientas combinatorias desarrolladas en [3], en particular del concepto de *imagen dividida*, y de un corolario que se desprende de la *Formula de Fan* [14], el cual es la generalización del *Lema del Indice* presentado en [16].

Uno de los principales resultados obtenidos, es una caracterización combinatoria del número de ejecuciones en las que un algoritmo renombra los procesadores haciendo uso únicamente de nombres pares o impares del conjunto de nombres de salida. En otras palabras, presentamos una caracterización combinatoria del número de *simplejos monocromáticos* en un complejo que tiene ciertas características de simetría en cuanto *estructura y coloración* en la *frontera*. Este resultado por sí solo es interesante, ya que muestra la forma en que se pueden aplicar algunas técnicas de topología combinatoria, como el Lema del Indice o la Formula de Fan, para determinar ciertas propiedades al interior de un complejo sin tener más información que la estructura que guarda en la frontera. Aún más, todas estas técnicas capturan lo realizado por la maquinaria de topología algebraica que se utilizan en [18], en particular los mapeos entre cadenas de simplejos. Desde el punto de vista topológico, nuestra caracterización no es más que una descripción combinatoria del grado de un mapeo de una esfera a sí misma que tiene ciertas características de simetría, el cual es un tema de gran interés en el área.

Ayudados de nuestra caracterización, obtenemos una demostración totalmente combinatoria de imposibilidad del k -renombrado, $k < 2n$, para una infinidad de casos. También presentamos un complejo en dimensión 5 el cual demuestra directamente que el avance obtenido en [3] tiene un error. Pero además,

este complejo también implica que las demostraciones anteriores [18, 19] tienen un error. Por lo tanto, nuestra principal contribución es que presentamos la primera demostración de imposibilidad del k -renombrado, $k < 2n$, para ciertos casos. Todavía más, con todo lo obtenido podemos empezar a preguntarnos si realmente el renombrado con esa cantidad de nombres es imposible de solucionarse o bajo que condiciones es posible hacerlo.

1.3. Organización de la Tesis

La organización de la tesis es la siguiente. El capítulo 2 muestra la especificación detallada del problema del renombrado y el conocimiento que se tiene de él. También estudia brevemente el modelo de memoria compartida, así como un modelo equivalente que será el que ocuparemos en el resto de la tesis. Los conceptos topológicos necesarios para el modelado de problemas y algoritmos distribuidos son presentados en el capítulo 3. El capítulo 4 muestra nuestro resultado principal. Por último, nuestras conclusiones y el trabajo que resta por hacerse en el futuro aparecen en el capítulo 5.

Capítulo 2

El problema del Renombrado

Por sus fuertes implicaciones teóricas y prácticas, se considera al *renombrado* como un problema fundamental dentro del área de los sistemas distribuidos. El estudio de este y otros problemas mas han permitido desarrollar los fundamentos teóricos del cómputo distribuido.

En este capítulo presentamos un panorama del problema del renombrado; su especificación, algunos algoritmos y sus cotas. En la sección 2.1 estudiamos los modelos asíncronos de memoria compartida y envío de mensajes, así como otro modelo equivalente que a la postre será de gran importancia para nuestros resultados. En la sección 2.2 presentamos la especificación del problema y el conocimiento que se tiene de él. También mostramos dos algoritmos sencillos que lo solucionan.

2.1. Modelos de Cómputo

A continuación presentamos el modelo asíncrono de memoria compartida. También explicaremos, de forma muy breve, en que consiste el modelo asíncrono de envío de mensajes en el que menos de la mitad de los procesadores pueden fallar. La relación fundamental que guardan estos dos modelos es que son equivalentes [4]. A saber, si existe un algoritmo para un problema dado en cualquiera de los dos modelos, entonces también existe un algoritmo que soluciona el mismo problema en el otro modelo. El modelo asíncrono de memoria compartida será el que utilizaremos para estudiar el renombrado.

2.1.1. Memoria Compartida

El modelo que ocuparemos es estándar y ha sido utilizado en varios artículos. Seguiremos fuertemente la notación ocupada en [3]. Comenzaremos por definir en que consisten los procesadores del sistema para después describir la estructura de la memoria que ocupan para comunicarse.

Consideramos a un *procesador* como una máquina de estados determinista, la cual consiste en un conjunto, posiblemente infinito, de *estados locales* y dos subconjuntos disjuntos de este a los que llamamos conjunto de *estados iniciales* y conjunto de *estados finales*. De esta forma, un *sistema* consta de una colección de $n + 1$ procesadores identificados como $p_0 \dots p_n$. Tradicionalmente se consideran sistemas con n procesadores identificados como $p_1 \dots p_n$. En el siguiente capítulo será claro por qué cuando se estudian los sistemas distribuidos a través de la topología resulta útil definir un sistema de la forma en que lo hicimos.

Ahora bien, los procesadores del sistema se comunican a través de una *memoria compartida* que consta de $n + 1$ *registros atómicos*, $r_0 \dots r_{n+1}$, uno por cada procesador. El procesador p_i cuenta con las operaciones atómicas *read* y *write* sobre los registros. La operación $read_i(r_j)$ hace una lectura del registro r_j , mientras que la operación $write_i(v)$ escribe el valor v en el registro r_i , el registro que le pertenece. La figura 2.1 presenta un ejemplo de esto. Los registros cuentan con un dominio de valores que pueden almacenar. Dentro de este dominio existe el valor indefinido \perp , el cual no puede ser pasado como parámetro a la operación de escritura. Debido a que no estamos restringiendo en nada el tamaño de los registros, asumimos que r_i tiene capacidad suficiente para almacenar por completo el estado actual del p_i .

La *configuración* del sistema es el conjunto que consiste en los estados locales actuales de los procesadores y el contenido actual de los registros. De forma más precisa, es el vector $C = \langle s_0 \dots s_n, v_0 \dots v_n \rangle$, donde s_i es el estado local de p_i y v_i es el valor de r_i . Denotamos como $state_i(C)$ y $value_i(C)$ a los valores s_i y v_i , respectivamente. Una *configuración inicial* es aquella en la que el estado local de cada procesador es un estado inicial y el valor de cada registro es \perp . De forma similar, una *configuración final* es aquella en la que el estado local de cada procesador es un estado final. Dentro del sistema un *evento* consiste en que el procesador p_i ejecute la operación *write* o *read*, pero no ambas, realice algunos cálculos locales y cambie al siguiente estado local. Claramente, la siguiente configuración del sistema depende del resultado de las modificaciones que genere el evento.

Decimos que la máquina de estados p_i ejecuta el *algoritmo local* A_i . De esta forma, el estado siguiente de p_i es determinado por A_i en base al estado actual. Esto es, si p_i ejecuta la operación $read_i(r_j)$, entonces A_i determina el estado siguiente en base al estado local actual de p_i y el valor que haya leído de r_j . En cambio, si p_i ejecuta la operación $write_i(v)$, entonces el estado siguiente está únicamente en función del estado local actual de p_i . Asumiremos que todas las máquinas de estado son idénticas. Es decir, todos los procesadores ejecutan el mismo algoritmo local y, por tanto, el comportamiento de un procesador no depende de su identificador. Observe que esto no merma en nada el poder de cómputo del modelo, puesto que si es necesario que el comportamiento de un algoritmo local dependa del identificador del procesador, entonces el identificador puede ser codificado en la entrada del algoritmo. Definimos como un *algoritmo* al conjunto A de los algoritmos locales $A_0 \dots A_n$.

Una *ejecución* del sistema consiste en una secuencia alternada, posiblemente

infinita, de configuraciones y eventos, $C_0, e_0, C_1, e_1 \dots C_{k-1}, e_{k-1}, C_k \dots$, donde C_0 es una configuración inicial y la configuración C_k es el resultado de aplicar el evento e_{k-1} a C^{k-1} . Decimos que la secuencia $e_0, e_1 \dots$ es la *calendarización* de la ejecución, además que definimos la *vista* del procesador p_i en una ejecución α , $\alpha|i$, como la secuencia de los estados locales del procesador p_i en la ejecución. Esto es, $\alpha|i = state_i(C_0), state_i(C_1) \dots$.

Observe que debido a que estamos asumiendo que los registros ejecutan operaciones atómicas, si dos procesadores ejecutan de forma concurrente una operación sobre un registro, una de ella se realizará antes que la otra. Es decir, realmente se ejecutan en serie. Por lo tanto, podemos definir la calendarización de una ejecución como una secuencia de eventos, como lo hemos hecho, y aún así no restringir en nada el momento en que los procesadores ejecutan operaciones sobre los registros.

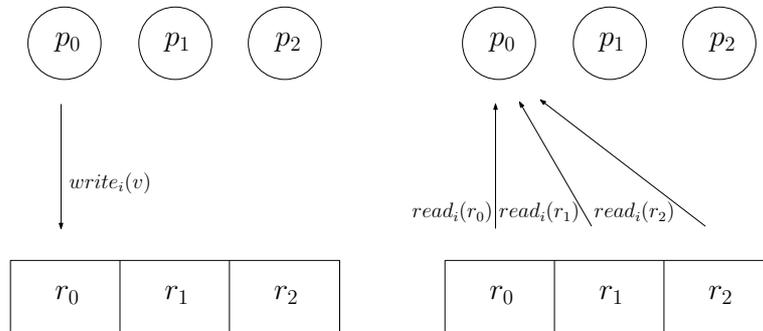


Figura 2.1: Memoria compartida.

Estamos interesados en modelar tareas de decisión (más adelante definiremos específicamente este concepto), por lo que agregamos a los procesadores un módulo de *entrada* y otro de *salida*. El módulo de entrada contiene la entrada al algoritmo local, el cual únicamente puede ser leído. Como mencionamos anteriormente, si es necesario que el comportamiento de un algoritmo local dependa del identificador del procesador, entonces es necesario pasar el identificador como parte de la entrada. El módulo de salida en un principio contiene el valor \perp , y una vez que un valor distinto ha sido escrito en él, ya no cambia. Cuando sucede esto decimos que el procesador ha *decidido*.

Dentro de un modelo de cómputo se pueden considerar diversos tipos de fallas de los procesadores. Inclusive podríamos aceptar fallas en el medio por el cual se comunican. Nosotros consideramos únicamente fallas de tipo *paro* de los procesadores. Esto es, si un procesador falla entonces ejecuta no más que un número finito de eventos antes de haber decidido. Decimos que un procesador es *correcto* si no ha fallado.

Normalmente el número de fallas que pueden ocurrir en un modelo se asocian a un parámetro t . A saber, el modelo asegura que al final de cualquier ejecución, por más catastrófica que sea, por lo menos $n - t + 1$ procesadores serán correctos.

Pueden existir situaciones en las que los procesadores locales de un algoritmo distribuido no deban retardar su decisión hasta recibir cierta información de algunos otros procesadores en el sistema. Posiblemente sistemas en los que sea muy importante la decisión oportuna de los procesadores. Por lo tanto, es de interés estudiar algoritmos libres de esperas, es decir, aquellos algoritmos que son tolerantes a cualquier número de fallas de los procesadores. Formalmente, decimos que un algoritmo es *wait-free* o *libre de espera* si en cualquier ejecución todo procesador decide o ejecuta un número finito de eventos. Esto implica que si un procesador ejecuta un número infinito de eventos, entonces decide en un número finito de ellos. Lo anterior tiene sentido en los casos en los que a pesar que un procesador ha decidido su salida, sigue ejecutando eventos con el fin de ayudar a los demás procesadores que no han decidido a que lo hagan. Cabe resaltar que para este caso el parámetro de tolerancia a fallos está tácitamente definido como $1 \leq t < n + 1$.

Más adelante será de gran importancia modelar la incapacidad que muestra un procesador para distinguir entre dos ejecuciones distintas. A saber, la vista del procesador es la misma para las dos ejecuciones. Decimos que las ejecuciones α_1 y α_2 son *indistinguibles* para el procesador p_i si $\alpha_1|i = \alpha_2|i$.

2.1.2. Envío de Mensajes

La definición de procesador en el modelo de envío de mensajes es básicamente la misma que la que dimos para el modelo de memoria compartida; es una máquina de estado determinista. La diferencia fundamental entre los dos modelos es que en el de envío de mensajes se cuenta con un medio de comunicación por el cual un procesador puede enviar paquetes de información, conocidos como *mensajes*, a cualquier otro procesador en el sistema, incluido él mismo. Por lo tanto, los procesadores tienen disponibles dos operaciones que les permiten enviar y recibir mensajes hacia y desde cualquier otro procesador. El sistema se puede describir como una gráfica completa en la que los nodos de la gráfica son los procesadores y las aristas el medio de comunicación.

No impondremos ninguna restricción en cuanto a las velocidades de los procesadores o el retardo de los mensajes en el medio de comunicación. Esto es, estamos considerando un sistema completamente asíncrono.

De forma similar que en el modelo de memoria compartida, consideramos únicamente fallas de los procesadores de tipo de paro. Para este caso el número de fallas es $t < (n + 1)/2$. Considerar un número mayor de fallas no tiene mucho sentido, ya que prácticamente ninguna tarea interesante tiene solución en este modelo cuando puedan fallar más de la mitad de los procesadores (por ejemplo [15]).

2.1.3. Tareas

Ahora definimos el concepto de tarea y cuándo decimos que un algoritmo la soluciona. Una *tarea* Δ consiste en un dominio de valores de entrada I y un dominio de valores de salida O , donde Δ especifica las salidas que los procesadores pueden decidir para cada combinación *válida* de los posibles valores de entrada que estos pueden tener. Por ejemplo, en la versión binaria del clásico problema del consenso cada procesador en el sistema tiene como entrada un valor binario que propondrá a los demás. El objetivo es que los procesadores decidan uno de los valores propuestos, de suerte tal que cualesquiera dos procesadores tienen el mismo valor de decisión. Para este caso tanto el dominio de valores de entrada I como el dominio de valores de salida O es $\{0, 1\}$. Observe que no existe ninguna restricción en cuanto a las posibles combinaciones de los valores de entrada a los procesadores, sin embargo, las posibles salidas están determinadas por las entradas de los procesadores. La Figura 2.2 muestra la especificación de la tarea del consenso binario para dos procesadores.

<i>Entradas</i>	Δ
(0, 0)	(0, 0)
(0, 1)	(0, 0); (1, 1)
(1, 0)	(0, 0); (1, 1)
(1, 1)	(1, 1)

Figura 2.2: Consenso binario.

De esta forma, decimos que un algoritmo, ya sea en el modelo de envío de mensajes o en el de memoria compartida, *soluciona* la tarea Δ , si cualquiera ejecución finita α del algoritmo puede ser extendida a una ejecución α' tal que los valores que deciden los procesadores son permitidos para las entradas en α . Observe que si en α algunos procesadores ya han decidido, entonces α' debe completar las decisiones de los demás procesadores de forma tal que los valores decididos sean permitidos para las entradas en α .

2.1.4. Ejecuciones *Immediate Snapshot*

Las ejecuciones *Immediate Snapshot* [10] son un subconjunto del conjunto que consiste en todas las ejecuciones posibles de un algoritmo que utiliza una memoria compartida como medio de comunicación. Este subconjunto se obtiene restringiendo las secuencias de operaciones que pueden ejecutar los procesadores sobre los registros de la memoria. De forma precisa, una ejecución *Immediate Snapshot* (IS) consta de una secuencia de *rondas*. Cada *ronda* consiste en un conjunto no vacío de identificadores de procesadores, los cuales denotan los procesadores activos en la ronda. Durante la ronda k -ésima los procesadores activos escriben uno a uno su estado completo, y después, ya que todos han terminado de escribir, hacen una lectura de toda la memoria, un *snapshot*. Es decir, cada ronda consta de una escritura concurrente seguida por una lectura concurrente.

El *número de pasos* de un procesador p_i en la ejecución α es el número de veces que está activo en α , mientras que el *número total de pasos* en α es la suma del número de pasos de todos los procesadores en α .

Debido a que las ejecuciones IS son un subconjunto del conjunto de todas las ejecuciones posibles, si un algoritmo soluciona un problema haciendo uso únicamente de ejecuciones IS, entonces también lo hace en el modelo general. La dirección contraria es la que parece en principio no ser cierta. Sin embargo, en [10] se demuestra que restringir un algoritmo a que utilice únicamente ejecuciones IS, no afecta en nada el poder de cómputo. Es decir, todo aquello que puede ser solucionado en el modelo libre de espera de memoria compartida también puede solucionarse con ejecuciones IS y viceversa.

Lema 2.1.1 ([3, Lema 3.1]) *Sea Δ una tarea. Existe un algoritmo libre de espera que soluciona Δ si y solo si existe un algoritmo libre de espera que soluciona Δ utilizando únicamente ejecuciones IS.*

En la figura 2.3 se observan un par de ejecuciones IS para las cuales únicamente el procesador p_0 puede distinguir entre ellas. En la ejecución α_1 el procesador p_1 únicamente puede ver el valor que él escribió en la memoria en la ronda 1, puesto que nadie más ha escrito en ella. Por lo tanto, p_1 únicamente se ve a él mismo. Siguiendo con esta idea, en la ronda 2 el procesador p_0 únicamente ve a p_1 y a él mismo, mientras que p_2 ve a todos los procesadores. En la ejecución α_2 los procesadores p_1 y p_2 tendrán la misma vista que en α_1 , sin embargo, p_0 verá todos los procesadores. Es decir, p_0 puede distinguir entre las ejecuciones α_1 y α_2 .

<i>Ronda :</i>	1	2	3		<i>Ronda :</i>	1	2	3
$p_0 :$		wr			$p_0 :$		wr	
$p_1 :$	wr				$p_1 :$	wr		
$p_2 :$			wr		$p_2 :$		wr	
		α_1					α_2	

Figura 2.3: Ejecuciones IS indistinguibles para p_1 y p_2 .

2.2. El Renombrado

En esta sección presentamos la especificación del problema del renombrado y las cotas que se conocen de él. También mostramos dos algoritmos sencillos que lo solucionan para dos diferentes dominios de nombres nuevos.

2.2.1. Especificación

En el problema del *renombrado* [5] cada procesador en el sistema comienza con un nombre tomado de dominio muy grande, posiblemente no acotado, de *nombres originales*, de suerte tal que cualesquiera dos nombres originales son distintos. El objetivo es simple, cada procesador correcto debe decidir de forma irreversible un nombre de un dominio de *nombres nuevos* lo más pequeño posible, de forma tal que se cumpla alguna de las siguientes condiciones:

1. *Unicidad*. Cualesquiera dos nombres nuevos son distintos.
2. *Preservación de orden*. Si $p < q$ entonces el nombre nuevo del procesador con nombre original p es menor que el nombre nuevo del procesador con nombre original q .

Cuando el dominio de nombres nuevos sea $[0 \dots k]$, nos referiremos al problema como el *k-renombrado*.

Advierta que debido a que los nombres originales de los procesadores son distintos, la segunda condición también implican que no existen dos procesadores con nombres nuevo iguales. En adelante nos ocuparemos de estudiar únicamente el problema del renombrado con la restricción de unicidad. Pero, ¿por qué motivo podría ser interesante estudiar el problema del renombrado?

En el estudio de los sistemas distribuidos es de interés conocer la cantidad o el tamaño de los mensajes que ocupa un algoritmo para cumplir su cometido. Para el caso del modelo de memoria compartida, sería el número de veces que es necesario escribir en los registros o el tamaño de los mismos. Por lo tanto, es de interés desarrollar algoritmos que solucionen el problema del renombrado haciendo uso de un dominio de nombres nuevos lo más pequeño posible, ya que estos pueden ocuparse para identificar los procesadores y así reducir el tamaño de los mensajes o registros ocupados por un algoritmo.

Aún más, un algoritmo que solucione el problema del renombrado puede servir para solucionar muchos otros problemas. Por ejemplo, suponga que se desea desarrollar un algoritmo para asignar de forma segura un conjunto de $k + 1$ recursos, digamos impresoras, a un conjunto de procesadores. Por supuesto que cada impresora deberá estar asignada a un solo procesador. El problema puede resolverse haciendo uso de un algoritmo de *k-renombrado* cada vez que un conjunto de procesadores desee ocupar las impresoras.

Sin embargo, existe una solución trivial al problema. Si el procesador p_i conociera su identificador entonces podría decidir el nombre nuevo i y de esta forma resolver el problema haciendo uso del dominio de nombres de salida más pequeño posible, $0 \dots n$. Para evitar este tipo de soluciones triviales, se requiere que los algoritmos sean *anónimos*. Es decir, el comportamiento de un procesador depende únicamente del nombre de entrada y no del identificador externo que tiene el procesador. En otras palabras, el procesador p_i con nombre

original x ejecuta el mismo algoritmo que el procesador p_j con nombre original x .

Hablando en términos del modelo de cómputo que estamos ocupando, un algoritmo es anónimo si cumple lo siguiente. Considere una ejecución E del algoritmo A , tal que el procesador p_i ha decidido el valor v . Decimos que A es anónimo si al intercambiar i por j , y viceversa, en E , la ejecución E' resultante es válida y además p_j decide v en E' . En el capítulo 4 daremos una definición formal de esto.

2.2.2. Cotas

Ahora presentamos un poco del conocimiento que se tiene acerca del renombrado, así como un problema abierto con respecto a su demostración de imposibilidad.

En [5] se presenta un algoritmo que soluciona $2n$ -renombrado en el modelo asíncrono de envío de mensajes en el que menos de la mayoría de los procesadores pueden fallar. De hecho, este algoritmo es *basado en comparaciones*; los procesadores únicamente pueden comparar sus nombres originales y, por tanto, ni siquiera saben cual es su nombre original. Allí también se demuestra que en ese modelo no existe algoritmo que solucione el $(n + 1)$ -renombrado. Dada la equivalencia entre modelos, los resultados obtenidos en [5] también son válidos para el modelo libre de espera de memoria compartida. Es decir, existe un algoritmo libre de espera para el $2n$ -renombrado, pero no lo existe para el $(n + 1)$ -renombrado. Sin embargo, nadie sabía que pasaba entre $n + 1$ y $2n$.

En el año de 1993 tres grupos de investigación [9, 18, 22] independientes demostraron la fuerte relación que existe entre los sistemas distribuidos y la topología. Estos trabajos dieron como resultado una caracterización topológica de lo que es computable *libre de espera* en el modelo asíncrono de memoria compartida. Ellos observaron que los *estados* de los procesadores al final de una ejecución de un algoritmo pueden ser representados por un *simplejo*, el cual no es más que un conjunto finito, mientras que todas las ejecuciones posibles del algoritmo pueden representarse como un *complejo*, un objeto matemático construido a base de simplejos. En el capítulo siguiente daremos la definición formal de estos conceptos.

El resultado fundamental en [9, 18, 22], es que los complejos de las ejecuciones de los algoritmos asíncronos libres de espera de memoria compartida son *sólidos*, es decir, no tienen hoyos. Esta invariante topológica es de gran importancia, ya que permitió responder la pregunta abierta acerca de la imposibilidad del renombrado.

En [18, 19] se demuestra que no existe algoritmo libre de espera que solucione el k -renombrado, $k < 2n$. Un aspecto fundamental en el resultado es que el conjunto de nombres nuevos $0 \dots k$, no tiene $n + 1$ nombres pares o impares. Básicamente lo que se demuestra es que para todo algoritmo libre de espera en el modelo de memoria compartida, siempre existe una ejecución en la que el algoritmo renombra los procesadores haciendo uso únicamente de nombres

pares o impares, lo cual es imposible con este rango.

Sin embargo, las demostraciones en [18, 19] se basan en técnicas de topología algebraica, lo cual las hace poco accesibles para la mayoría de los estudiosos de la computación, además que no es del todo claro cual es la razón fundamental del resultado. No obstante, se han realizado avances hacia una demostración combinatoria más simple [3], aunque la parte crucial de la misma sigue dependiendo de argumentos del área de la topología algebraica.

Este es precisamente el objetivo de los siguientes capítulos de la tesis: contribuir al desarrollo de una demostración combinatoria de imposibilidad del k -renombrado, $k < 2n$, que permita entender de forma más clara por qué siempre aparecen este tipo de ejecuciones imposibles en el modelo asíncrono libre de espera de memoria compartida.

2.2.3. Dos Algoritmos de Renombrado

En esta sección presentamos un par de algoritmos libres de espera basados en comparaciones que solucionan el renombrado para tres procesadores utilizando los dominios de nombres nuevos $\{0, 1, 2, 3, 4, 5\}$ y $\{0, 1, 2, 3, 4\}$, respectivamente. Estos algoritmos se restringen a ocupar ejecuciones IS y cumplen la condición de anonimato.

Un Primer Intento

Primero presentamos un algoritmo que utiliza el dominio de nombres nuevos $\{0, 1, 2, 3, 4, 5\}$. La idea del algoritmo es la siguiente. Los nombres nuevos se dividen en tres conjuntos; el primero de ellos contiene el nombre 0, el segundo a 1 y 2, y el tercero a 3, 4 y 5. Un procesador p ejecuta la operación de escritura y *snapshot* sobre la memoria y de acuerdo al número de procesadores que ve en este último selecciona el conjunto que ocupará para decidir su nombre nuevo. Esto es, si p ve un solo procesador (nadie ha escrito en la memoria mas que él), entonces escogerá el primer conjunto, si ve dos procesadores, entonces escogerá el segundo conjunto y si ve tres procesadores, entonces escogerá el tercer conjunto. Después de esto, p decide el nombre en el conjunto que tiene el mismo orden que el orden que guarda su nombre original con respecto al nombre original de los demás procesadores que vio en el *snapshot*. Por ejemplo, si su nombre original es menor al de todos los demás, entonces decidirá el nombre de menor valor en el conjunto.

Denotamos como $snapshot(i)$ al conjunto que consiste en los registros de aquellos procesadores que han estado activos exactamente i veces en la lectura *snapshot*. De esta forma, $|snapshot(i)|$ es el número de procesadores que están activos i veces en *snapshot*. El algoritmo es el siguiente.

```
AlgRenombrado1(nomOrig)  
snapshot ← ejecuta wr en la memoria  
if  $|snapshot(1)| = 1$  then  
  nomNuevo ← 0
```

```

if |snapshot(1)| = 2 then
  if nomOrig es menor al otro nombre en snapshot(1) then
    nomNuevo ← 1
  else
    nomNuevo ← 2
if |snapshot(1)| = 3 then
  if nomOrig es menor a los otros dos nombres en snapshot(1) then
    nomNuevo ← 3
  else if nomOrig es mayor a los otros dos nombres en snapshot(1) then
    nomNuevo ← 5
  else
    nomNuevo ← 4
decide nomNuevo

```

Observe que cada nombre original en *snapshot* tiene bien definido cual es el nombre que le corresponde en el conjunto de nombres nuevos y, además, un nombre nuevo pertenece únicamente a un conjunto. Puesto que los procesadores tienen nombres originales distintos, cada procesador decide un nombre nuevo diferente.

Este algoritmo se puede extender fácilmente para un número arbitrario de procesadores. Basta con utilizar un conjunto con i nombres nuevos para cuando un procesador ve i procesadores en *snapshot*. No obstante, la solución es muy costosa en cuanto a nombres, ya que para n procesadores el algoritmo necesita un dominio de nombres nuevos con $\frac{n(n+1)}{2}$ nombres.

Eliminando un Nombre

Ahora presentamos un algoritmo, derivado del que se mostró en la sección anterior, que ocupa el rango de nombres nuevos $\{0, 1, 2, 3, 4\}$. Esto nos da la solución óptima para tres procesadores (ver sección 2.2.2). La diferencia principal es que en este, un procesador ejecuta una segunda operación de escritura y *snapshot* cuándo ve tres procesadores en la primera lectura de la memoria.

```

AlgRenombrado2(nomOrig)
snapshot1 ← ejecuta wr en la memoria
if |snapshot1(1)| = 1 then
  nomNuevo ← 0
if |snapshot1(1)| = 2 then
  if nomOrig es menor al otro nombre en snapshot1(1) then
    nomNuevo ← 1
  else
    nomNuevo ← 2
if |snapshot1(1)| = 3 then
  snapshot2 ← ejecuta wr en la memoria
  if |snapshot2(2)| = 1 then
    nomNuevo ← 4
  if |snapshot2(2)| = 2 or |snapshot2(2)| = 3 then
    if nomOrig es menor a los otros dos nombres en snapshot1(1) then
      nomNuevo ← 1

```

```

else if nomOrig es mayor a los otros dos nombres en snapshot1(1) then
    nomNuevo ← 3
else
    nomNuevo ← 2
decide nomNuevo

```

El truco consiste en lo siguiente: cuando un procesador p toma un segundo *snapshot*, sabe de antemano que los nombres 3 y 4 no ha sido ocupados. Advierta que puede ser el caso, no necesariamente, que 0, 1 o 2 ya han sido utilizados, sin embargo, p no puede tener certeza de esto. De esta forma, si p ve en su segundo *snapshot* que solo él ha estado activo dos veces, entonces lo único seguro que puede decidir es 3 o 4. Supongamos que decide 4. No sucede así cuando p ve en su segundo *snapshot* que dos o tres procesadores han estado activos dos veces. De hecho en estos casos tiene más información. En este punto puede asegurar que los nombres 1, 2 y 3 no han sido utilizados y decidir alguno de ellos. Si por lo menos dos procesadores han estado activos dos veces, no es posible que los nombres 1 y 2 estén ocupados.

Utilizando un truco similar podemos extender el algoritmo anterior para cualquier número de procesadores. En [10] se presenta un algoritmo que soluciona el $2n$ -renombrado haciendo uso de ejecuciones IS.

Capítulo 3

Los Sistemas Distribuidos y la Topología

En el año de 1993 tres grupos de investigación [9, 18, 22] independientes demostraron la fuerte relación que existe entre los sistemas distribuidos y la topología. Este es un resultado fundamental para los sistemas distribuidos, ya que ha permitido responder importantes preguntas abiertas en el área. En éste capítulo estudiamos esta relación. Un primer acercamiento a esta importante área de las matemáticas, la topología, puede obtenerse en [1, 16, 21].

En la sección 3.1 presentamos los conceptos topológicos combinatorios que ocuparemos a lo largo de este y el siguiente capítulo. En particular, en las secciones 3.1.3 y 3.1.4 introduciremos las principales herramientas combinatorias que ocuparemos. A saber, el lema del índice y las imágenes divididas. A lo largo de la sección 3.1.4 también probamos varias propiedades de las imágenes divididas que nos resultarán útiles más adelante. En la sección 3.2 estudiamos la forma en que se modelan tareas y algoritmos haciendo uso de la topología, en particular haciendo uso de las imágenes divididas.

3.1. Conceptos Topológicos

Comenzamos presentando los conceptos básicos con los que definiremos las herramientas topológicas combinatorias que utilizaremos para el estudio de los sistemas distribuidos.

3.1.1. Conceptos Básicos

Un *polígono* es una figura plana acotada por un camino cerrado formado por líneas rectas a las cuales llamamos *aristas*. Los *vértices* o *esquinas* del polígono son los puntos donde se intersectan sus aristas. De esta forma, un *polígono simple* es aquel en el que sus aristas únicamente se intersectan en los vértices del polígono. Una *triangulación* es una subdivisión de un polígono simple de forma

tal que toda arista pertenece a uno o dos triángulos.

Un *complejo* K es un conjunto de conjuntos finitos no vacíos, cerrado bajo la operación de contención. Es decir, si σ es un elemento de K entonces K contiene a todo subconjunto no vacío de σ . A todo elemento de un complejo K le llamamos *simplejo*. La dimensión de un simplejo σ , $\dim(\sigma)$, es el número de sus elementos menos 1. Si σ tiene $n + 1$ elementos entonces decimos que es un n -*simplejo* y lo denotamos como σ^n . Decimos que τ es una *cara* o $\dim(\tau)$ -*cara* de σ si τ es un subconjunto no vacío de σ . Si τ es diferente de σ entonces decimos que τ es una *cara propia* de σ . Observe que el número de caras de dimensión i , i -*caras*, de un n -simplejo está dado por $\binom{n+1}{i+1}$, $i \leq n$. La dimensión de un complejo K es la máxima dimensión de entre las dimensiones de sus simplejos. Además, decimos que un complejo K de dimensión n es un n -*complejo* y lo denotamos como K^n . Un complejo L es un *subcomplejo* del complejo K si $L \subseteq K$.

Gráficamente representamos a un 0-simplejo como un solo punto, a un 1-simplejo como una línea, a un 2-simplejo como un triángulo y a un 3-simplejo como un tetraedro (ver figura 1.2). En la Figura 3.1 podemos ver dos complejos de dimensión 1 y 2, respectivamente.

Sean los complejos K y L y un mapeo f que va de los vértices de K a los vértices de L . Decimos que el mapeo f es *simplicial* si para todo simplejo $\sigma = \{v_0, v_1 \dots v_{n-1}, v_n\}$ de K , el conjunto $\tau = \{f(v_0), f(v_1) \dots f(v_{n-1}), f(v_n)\}$ es un simplejo de L , posiblemente con dimensión $\dim(\tau) < \dim(\sigma)$. Esto último implica que f no necesariamente debe ser uno a uno y, por tanto, que dos simplejos diferentes en K pueden ser mapeados a un mismo simplejo en L . Algunas veces resulta más fácil para la escritura extender el mapeo f a simplejos de K : un simplejo σ de K es mapeado al simplejo $f(\sigma)$ de L . Una idea similar se puede ocupar para subconjuntos de K .

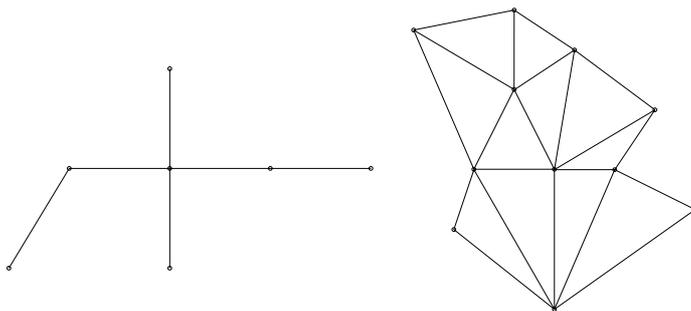


Figura 3.1: Complejos de dimensión 1 y 2.

La j -*gráfica* del complejo K^n , $1 \leq j \leq n$, es la gráfica que consiste en un vértice por cada j -simplejo de K^n y una arista entre dos vértices si los simplejos que representan dichos vértices comparten una $(j - 1)$ -cara. Decimos que K^n es j -*conexo* si la j -gráfica de K^n es conexa. Un j -*camino* en K^n es una secuencia de j -simplejos de K^n tales que cualesquiera dos j -simplejos consecutivos en la

secuencia comparten una $(j - 1)$ -cara. Por supuesto que todo j -camino en K^n tiene un camino equivalente en la j -gráfica de K^n .

Un complejo K^n es *completo* en dimensión i , $i \leq n$, si todo j -simplejo, $j \leq i$, de K^n es cara de por lo menos un i -simplejo de K^n . Si el complejo K^n es completo para dimensión n , entonces decimos que un $(n - 1)$ -simplejo σ de K^n es *externo* si es cara de exactamente un n -simplejo de K^n . De otra forma decimos que σ es *interno*. Con esto definimos la frontera de K^n , $bd(K^n)$, como el subcomplejo con todos los simplejos *externos* y todas las caras de cada uno de estos. De forma similar, un vértice en K^n es *externo* si pertenece a algún simplejo de $bd(K^n)$. De otra forma decimos que es *interno*. Por ejemplo, la frontera del 1-complejo en la Figura 3.1 son los vértices en los extremos de las líneas que lo forman, mientras que la frontera del 2-complejo que se aprecia en la misma figura es el contorno del polígono.

Una *coloración* de un complejo K^n es una función c que va de los vértices de K^n a un conjunto C que llamamos *colores*. Considere un simplejo $\tau \in K^n$. Denotamos como $c(\tau)$ al conjunto de colores que asigna c a los vértices de τ . Si $|C| = 2$ entonces decimos que c es una coloración binaria. En lo subsecuente, asumiremos que toda coloración binaria ocupa los colores $\{0, 1\}$. Frecuentemente usaremos el conjunto $ID^n = \{0, 1 \dots n\}$ como colores. Denotaremos al subconjunto $ID^n - I$ como ID_I^n . Si $I = \{x\}$, simplemente escribimos ID_x^n . Decimos que la coloración c de un simplejo σ es *propia* si cualesquiera dos vértices diferentes de σ tienen colores diferentes. Observe que si $|c(\sigma)| = dim(\sigma) + 1$ entonces σ está coloreado propiamente bajo c . Frecuentemente ocuparemos esta notación. Si todos los vértices de σ tiene el mismo color b entonces decimos que σ es *b-monocromático* o simplemente *monocromático*. Considere un simplejo σ^n con una coloración propia c . Denotamos como σ_i^{n-1} a la $(n - 1)$ -cara de σ^n que no tiene el vértice coloreado i bajo c . Ahora considere dos complejos coloreados K y L . Un mapeo simplicial f entre K y L *preserva colores* si para todo vértice v , $f(v)$ tiene el mismo color que v .

Considere un conjunto C con $n + 1$ elementos. Una *secuencia* S de C es una lista ordenada de los elementos de C . Denotamos como S_i al i -ésimo elemento de S , donde $0 \leq i \leq n$. Una *transposición* de S consiste en intercambiar de posición a dos elementos de S . Si estos son el mismo, entonces decimos que la transposición es la *identidad*. Una secuencia S' de C es una *transposición par* de otra S si podemos obtener S' aplicando un número par de transposiciones diferentes a la transposición identidad sobre S . De forma similar definimos una *transposición impar*.

Ahora considere un subconjunto I de ID^n . Sea $S = \langle S_0, S_1 \dots S_{|I|-1} \rangle$ la secuencia de los elementos de I en orden ascendente. El *rank* de un elemento x de I es la posición que guarda en S . Es decir, $rank(x, I) = j$ si y solo si $S_j = x$. Si no existe ambigüedad simplemente escribiremos $rank(x)$.

Decimos que una función f de un conjunto C a él mismo es una *permutación* de C si es uno a uno. Si además para todo $x \in C$, $f(x) = x$ entonces f es la *identidad* de C .

3.1.2. Pseudovariedades

Con lo visto en la sección anterior podemos hacer una definición exacta del termino *pseudovariedad*. Sea K^n un complejo completo en dimensión n . Decimos que K^n es una n -*pseudovariedad* si todo $(n-1)$ -simplejo de K^n es contenido por uno o dos n -simplejos. Si todo $(n-1)$ -simplejo es contenido por exactamente dos n -simplejos, entonces decimos que K^n es una n -*variedad*. Por supuesto que una n -variedad es una n -pseudovariedad en la que todos los $(n-1)$ -simplejos están contenidos en dos n -simplejos. Considere un vértice v de K^n . El *complejo estrella* de v en K^n , $st(v, K^n)$, es el complejo que consiste en todos los n -simplejos de K^n que contienen a v , junto con todas sus caras. El *complejo enlace* de v en K^n , $lk(v, K^n)$, es $bd(st(v, K^n))$. Cuando no exista ambigüedad simplemente escribiremos $st(v)$ o $lk(v)$.

Sea σ un n -simplejo. Un complejo en el que pondremos especial atención en las siguientes secciones y capítulos es el complejo que contiene a σ junto con todas sus caras, al cual denotaremos como $M(\sigma)$. Es fácil ver que $M(\sigma)$ es una pseudovariedad.

Una superficie es orientable si puede deslizarse por ella un disco con una rotación definida, ya sea en el sentido de las manecillas del reloj o a la inversa, y al regresar al punto de partida el disco tendrá la misma rotación, sin importar el trayecto que se haya seguido. Por su puesto que no todos las superficies son orientables; una banda de Möbius es un ejemplo de esto. Este concepto de orientabilidad también se puede aplicar a las pseudovariedades que hemos definido. Antes presentar el concepto de orientación sobre una pseudovariedad, primero definimos a que se refiere la orientación de un solo simplejo.

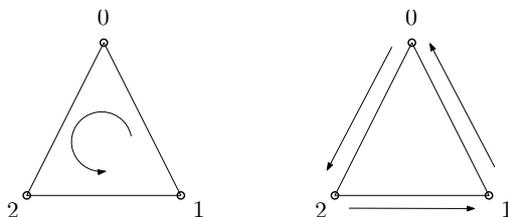


Figura 3.2: Un simplejo orientado.

Considere el simplejo $\sigma^n = \{v_0, v_1 \dots v_n\}$, $n > 0$. Una *orientación* de σ^n es un conjunto el cual consiste en una secuencia de los vértices de σ^n y todas las transposiciones pares de ella. Existen exactamente dos posibles orientaciones de σ^n . Denotamos como $d = +1$ a la orientación de σ^n que contiene la secuencia $\langle v_0, v_1 \dots v_n \rangle$, y como $d = -1$ la otra orientación. Por lo tanto, esta última contiene la secuencia $\langle v_1, v_0 \dots v_n \rangle$. Para $n = 0$, existe una única forma de ordenar el único vértice de un 0-simplejo y, por tanto, tiene únicamente una orientación. No obstante, podemos asociar a ella $+1$ o -1 . Por lo tanto, podemos asumir que un 0-simplejo también tiene dos orientaciones. Una orientación d de σ^n , $n > 0$, induce una orientación a todas sus $(n-1)$ -caras: la cara σ_i^{n-1} adquiere la orientación $(-1)^i d$, donde σ_i^{n-1} es la $(n-1)$ -cara de σ^n sin el vértice v_i . Si σ^n

tiene una coloración propia id con ID^n , entonces denotamos como $d = +1$ a la orientación de σ^n que contiene la secuencia $\langle 0, 1 \dots n \rangle$, y como $d = -1$ la otra orientación. De forma similar, σ_i^{n-1} adquiere la orientación $(-1)^i d$, donde σ_i^{n-1} es la $(n-1)$ -cara de σ^n sin el vértice coloreado i bajo id . La figura 3.2 muestra un 2-simplejo orientado y las orientaciones que induce a sus 1-caras.

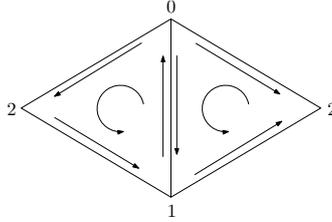


Figura 3.3: Una pseudovariiedad orientada coherentemente.

Sea σ un n -simplejo con una coloración propia id con ID^n . Considere un par de secuencias S y T de ID^n . Sea S' la secuencia de σ^n inducida por S con respecto a id . Esto es, S' es la secuencia tal que $id(S'_i) = S_i$, $0 \leq i \leq n$. Decimos que la secuencia S induce la orientación S' a σ con respecto a id . Ahora, sea T' la secuencia de σ inducida por T con respecto a alguna coloración propia f de σ (por supuesto que f debe usar los colores ID^n). La secuencia T' pertenece a la orientación de σ con respecto a f si T' pertenece a la orientación de σ .

Una pseudovariiedad K^n es *orientable* si es posible asignar una orientación a cada uno de sus n -simplejos de forma tal que si dos n -simplejos σ_1 y σ_2 comparten una $(n-1)$ -cara τ^{n-1} , entonces τ^{n-1} adquiere orientaciones opuestas de σ_1 y σ_2 . Si K^n tiene una orientación como esta, decimos que tiene una *orientación coherente* o simplemente que está *orientada coherentemente*.

Una pseudovariiedad K^n es *cromática* si tiene una coloración f con $n+1$ colores tal que todo n -simplejo de K^n está coloreado propiamente bajo f . El siguiente lema da una condición necesaria y suficiente de una pseudovariiedad cromática orientable.

Lema 3.1.1 ([3, Lema 5.12]) *Una pseudovariiedad cromática K^n es orientable si y solo si sus n -simplejos pueden ser particionados en dos clases disjuntas tales que si dos n -simplejos comparten una $(n-1)$ -cara entonces pertenecen a clases diferentes.*

3.1.3. El Lema del Índice

El Lema del Índice es una de las herramientas por medio de las cuales obtendremos en el capítulo 4 una caracterización combinatoria del número de simplejos monocromáticos en el complejo de ejecuciones de un algoritmo del renombrado. Primero presentamos el caso en dimensión 2 con el fin de tener la

intuición de lo que trata el lema.

Lema del Índice para Dimensión 2

Sea T una triangulación de un polígono simple en la que cada vértice está coloreado con 0, 1 o 2. En T pueden haber distintos tipos de triángulos en cuanto a su coloración. Decimos que un triángulo en T está *coloreado propiamente* si tiene tres colores distintos asociados a sus vértices. Ahora bien, asumimos que un triángulo coloreado propiamente tiene orientación $+1$ si al leer los colores 0, 1, 2, en ese orden, se sigue el sentido inverso a las manecillas del reloj. En otro caso, se sigue el sentido de las manecillas del reloj, el triángulo tiene orientación -1 . Asociamos dos cantidades a la triangulación T .

1. El *contenido* de T , C , es el número de triángulos coloreados propiamente contados por orientación. Es decir, cada triángulo coloreado propiamente con orientación d es contado como d .
2. El *índice* de T , I , es el número de aristas 01 (con los colores 0 y 1 en sus vértices, respectivamente) sobre la frontera contadas por orientación. Es decir, una arista 01 es contada como $+1$ si al leer sus colores 0, 1, en ese orden, alrededor del polígono se sigue el sentido inverso a las manecillas del reloj. De otra forma, se sigue el sentido de las manecillas del reloj, se cuenta como -1 .

Una prueba simple del siguiente lema se presenta en [16, pp. 46-47].

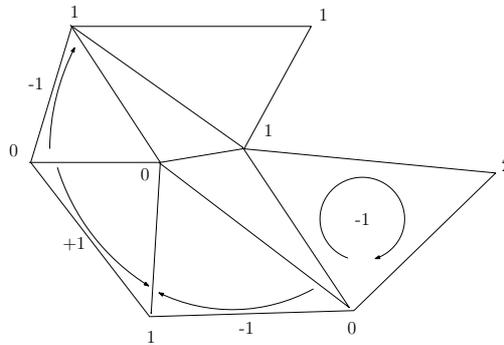


Figura 3.4: El Lema del Índice en acción.

Lema 3.1.2 (Lema del Índice [16, pp. 46-47]) *Sea una triangulación de un polígono simple en la que cada vértice está coloreado con 0, 1 o 2. Tenemos que $C = I$.*

La Figura 3.4 muestra un ejemplo del Lema del Índice. El único triángulo coloreado propiamente tienen orientación -1 y, por tanto, es contado por el contenido como -1 . En la frontera, una arista 01 es contada por el índice como $+1$ y las otras dos como -1 . Por lo tanto, el índice también es igual a -1 .

Intuitivamente podemos interpretar el índice de la siguiente manera. Considere un triángulo σ coloreado propiamente con $0, 1$ y 2 . Los colores $0, 1, 2$ de la triangulación T representan un mapeo simplicial de T a σ . De esta forma, podemos pensar el índice de T como el número de veces que la frontera de T envuelve la frontera de σ . Por ejemplo, en la Figura 3.4 de forma muy sencilla podemos verificar que la frontera de la triangulación se envuelve -1 vez en la frontera de un triángulo coloreado propiamente.

Lema del Índice para Dimensión n

Ahora presentamos la generalización del Lema del Índice, así como un lema que muestra la relación entre los diferentes índices que se pueden calcular sobre una pseudovariiedad coloreada. Estos dos se obtienen directamente de la *Formula de Fan* [14].

Sea K^n una pseudovariiedad orientada coherentemente. En la siguiente definición contamos un n -simplejo de K^n coloreado propiamente bajo f , por orientación. Es decir, el n -simplejo es contado como $+1$ si la secuencia $\langle 0, 1 \dots n \rangle$ pertenece a su orientación con respecto a f , y es contado como -1 en cualquier otro caso. De forma similar se cuenta un $(n - 1)$ -simplejo en la frontera, con la salvedad de utilizar la orientación que le induce el único n -simplejo de K^n que lo contiene.

Definición 3.1.3 (Índice y Contenido) *Para una coloración f de K^n con ID^n , sean*

1. *El contenido de K^n , $C(K^n)$, con respecto a f es el número de n -simplejos de K^n coloreados propiamente bajo f contados por orientación con respecto a f .*
2. *El índice de K^n , $I(K^n)$, con respecto a f es el número de $(n - 1)$ -simplejos de $bd(K^n)$ coloreados propiamente bajo f con $ID_n^n = \{0, \dots, n - 1\}$ contados por orientación con respecto a f .*

Si no existe ambigüedad simplemente escribiremos C^n o I^n . El siguiente lema es la generalización del Lema del Índice presentado en la sección anterior. Esta generalización puede ser obtenida fácilmente del Corolario 2 en [14].

Lema 3.1.4 (Lema del Índice para Dimensión n) *Sea K^n una pseudovariiedad orientada coherentemente que además está coloreada con ID^n . Entonces $C^n = (-1)^n I^n$.*

Sin embargo, el índice puede ser calculado con cualquier subconjunto de ID^n con n colores. Definimos como $I_i(K^n)$, I_i^n , al índice de K^n calculado con los colores ID_i^n . Observe que en el lema anterior $I^n = I_n^n$. El siguiente lema es el Corolario 2 en [14] reescrito usando la notación que hemos presentado hasta ahora.

Lema 3.1.5 (Generalización del Lema del Índice) *Sea K^n una pseudovariiedad orientada coherentemente que además está coloreada con ID^n . Entonces $C^n = (-1)^i I_i^n$.*

Un ejemplo del lema anterior se puede observar en la Figura 3.4. Observe que $I_2^2 = -1$ puesto que ya habíamos calculado $I = -1$. Además, los colores 0, 2, en ese orden, de la única arista 02 en la frontera se leen alrededor del polígono en sentido inverso a las manecillas del reloj. Por lo tanto, $I_1^2 = +1$. Advierta que $(-1)^2 I_2^2 = (-1)^1 I_1^2$.

De forma similar que como en el caso para dimensión 2, la coloración f en la Generalización del Lema del Índice 3.1.5 representa un mapeo simplicial hacia un n -simplejo σ coloreado propiamente con ID^n . De nueva cuenta, podemos interpretar el índice de K^n como el número de veces que $bd(K^n)$ se envuelve en $bd(\sigma)$.

Considere una pseudovariiedad K^n como la que asume la Generalización del Lema del Índice 3.1.5. Sea L^{n-1} el complejo que consiste en todos los $(n-1)$ -simplejos en $bd(K^n)$ que no tienen el color n , junto con todas sus caras. Es decir, si $\tau^{n-1} \in L^{n-1}$, entonces todo vértice de τ^{n-1} está coloreado por f con un color diferente a n . Además, asumimos que L^{n-1} tiene la orientación inducida por K^n . El siguiente lema muestra la relación recursiva que tienen los índices de K^n y L^{n-1} .

Lema 3.1.6 *Si L^{n-1} es una pseudovariiedad orientada coherentemente, entonces $I_n(K^n) = C(L^{n-1})$. Aún más, $I_n(K^n) = (-1)^j I_j(L^{n-1})$.*

Demostración: Primero observe que $I_n(K^n)$ cuenta todos los $(n-1)$ -simplejos de $bd(K^n)$ coloreados propiamente con ID_n^n , y el complejo L^{n-1} contiene todos los $(n-1)$ -simplejos de $bd(K^n)$ coloreados propiamente con ID_n^n . Por hipótesis, L^{n-1} es una pseudovariiedad orientada coherentemente que además tiene una coloración con $ID_n^n = \{0 \dots n-1\}$. Por lo tanto, por el la Generalización del Lema del Índice 3.1.5, $C(L^{n-1}) = (-1)^j I_j(L^{n-1})$. Observe que $I_n(K^n)$ y $C(L^{n-1})$ se calculan utilizando la misma orientación. Aún más, por la definición de índice y contenido, Definición 3.1.3, $I_n(K^n) = C(L^{n-1})$. Por lo tanto, $I_n(K^n) = (-1)^j I_j(L^{n-1})$.

□

3.1.4. Imágenes Divididas

Las imágenes divididas fueron introducidas en [3] con el objetivo de contar con herramientas combinatorias para el modelado de tareas y algoritmos distribuidos, así como obtener una demostración combinatoria del problema del renombrado. Estas jugarán un papel importantísimo en el desarrollo de nuestro resultado principal en el capítulo 4.

El rol de las imágenes divididas dentro de la relación entre los sistemas distribuidos y la topología es, a groso modo, el siguiente. Supongamos que tenemos un complejo I que representa todas las entradas posibles de un algoritmo, y otro mas E que representa todas las ejecuciones posibles del mismo (ejecuciones en las que los procesadores deciden). Una imagen dividida toma un subcomplejo I' de I y lo mapea a un subcomplejo de E . Intuitivamente, el mapeo representa las ejecuciones a las que puede llegar el algoritmo cuando los procesadores comienzan con las entradas en I' .

Definición y Propiedades Básicas

En esta sección presentamos la definición de las imágenes divididas, así como algunas de sus propiedades más elementales.

Definición 3.1.7 ([3, Definición 4.1]) Sean los complejos K^n y L^n y una función ψ que asigna a cada simplejo en L^n un subcomplejo finito de K^n . El complejo K^n es una imagen dividida de L^n bajo ψ si

1. $\psi(\emptyset) = \emptyset$
2. para todo $\tau \in K^n$ existe un simplejo $\sigma \in L^n$ tal que $\tau \in \psi(\sigma)$
3. para todo $\sigma^0 \in L^n$, $\psi(\sigma^0)$ es un vértice
4. para todo par $\sigma_1, \sigma_2 \in L^n$, $\psi(\sigma_1 \cap \sigma_2) = \psi(\sigma_1) \cap \psi(\sigma_2)$
5. para todo $\sigma \in L^n$, $\psi(\sigma)$ es una $\dim(\sigma)$ -pseudovariación con $bd(\psi(\sigma)) = \psi(bd(\sigma))$

El complejo K^n es una imagen dividida de L^n si existe alguna ψ para la cual K^n es una imagen dividida de L^n bajo ψ .

En la Figura 3.5 se puede ver un ejemplo de una imagen dividida. Cada triángulo (2-simplejo) de L^n es mapeado a una triangulación de un triángulo (2-pseudovariación) de K^n . Intuitivamente, la imagen dividida son las instrucciones de como subdividir L^n para obtener K^n .

Lema 3.1.8 ([3, Lema 4.2]) Sea K^n una imagen dividida de L^n bajo ψ .

1. para todo par $\sigma, \sigma' \in L^n$ si $\sigma' \subseteq \sigma$ entonces $\psi(\sigma') \subseteq \psi(\sigma)$

2. para todo par de j -simplejos $\sigma_1^j, \sigma_2^j \in L^n$, si $\sigma_1^j \neq \sigma_2^j$ y $\sigma_1^j \cap \sigma_2^j \neq \emptyset$ entonces $\psi(\sigma_1^j) \cap \psi(\sigma_2^j)$ es una pseudovariiedad de dimensi3n estrictamente menor que j , i.e., $|\sigma_1^j \cap \sigma_2^j| - 1$
3. para todo i -simplejo $\sigma^i \in L^n$, $\psi(\sigma^i)$ es una imagen dividida de $M(\sigma^i)$ bajo $\psi|_{M(\sigma^i)}$.
4. un simplejo $\tau^{n-1} \in K^n$ es externo si y solo si para alg3n simplejo externo $\sigma^{n-1} \in L^n$, $\tau^{n-1} \in \psi(\sigma^{n-1})$.

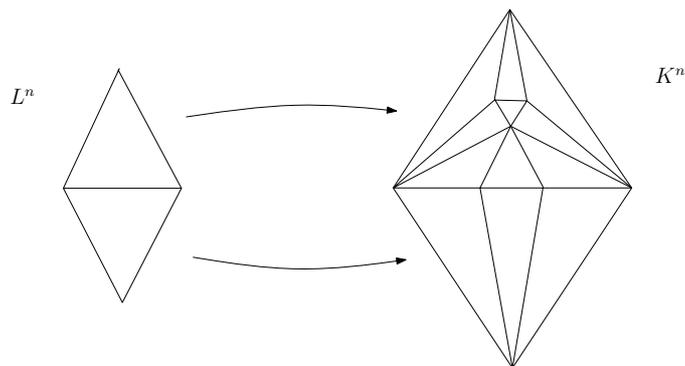


Figura 3.5: Ejemplo de una imagen dividida.

El siguiente lema muestra un tipo importante de im3genes divididas que ocuparemos en el siguiente cap3tulo.

Lema 3.1.9 ([3, Lema 4.3]) Sea K^n una imagen dividida de $M(\sigma^n)$ bajo ψ . Tenemos que $\psi(\sigma^n) = K^n$.

Con el objetivo de tener una notaci3n m3s sencilla, en lo subsecuente simplemente diremos que K^n es imagen dividida de σ^n bajo ψ . De hecho, en lo que resta de la tesis nos concentraremos en este tipo de im3genes divididas.

Los conceptos de conexidad y orientabilidad tambi3n pueden ser aplicados a las im3genes divididas. A saber, sea K^n una imagen dividida de σ^n bajo ψ . Decimos que la imagen dividida K^n es *conexa* si para toda cara σ^i de σ^n , si $i \geq 1$ entonces $\psi(\sigma^i)$ es conexa y si $i \geq 2$ entonces $bd(\psi(\sigma^i))$ es conexa. De forma similar, K^n es *orientable* si $\psi(\sigma^i)$ es orientable. Adem3s, decimos que K^n est3 *orientada coherentemente* si $\psi(\sigma^n)$ est3 orientada coherentemente.

Considere una imagen dividida K^n de σ^n bajo ψ . El *carrier* de un simplejo $\tau \in K^n$, $carr(\tau)$, es la cara σ de σ^n de menor dimensi3n tal que $\tau \in \psi(\sigma)$. Ahora considere un v3rtice $v \in K^n$; decimos que v tiene un *carrier de dimensi3n i* si $dim(carr(\{v\})) = i$. Viendo el Lema 3.1.8 (3, 4) podemos afirmar lo siguiente.

Afirmación 3.1.10 Considere una m -cara σ de σ^n . Un vértice $v \in \psi(\sigma)$ es un vértice interno de $\psi(\sigma)$ si y solo si tiene un carrier de dimensión m y, además, $\text{carr}(\{v\}) = \sigma$.

Ahora introducimos el concepto de *arista de cruce* en una imagen dividida. Sea K^n una imagen dividida de σ^n bajo ψ . Una *arista de cruce* de K^n es un 1-simplejo $\{u, v\} \in \text{bd}(K^n)$, tal que $u \in \psi(\sigma_1)$ y $v \in \psi(\sigma_2)$, donde σ_1 y σ_2 son m -caras diferentes de σ^n , $0 \leq m \leq n - 2$.

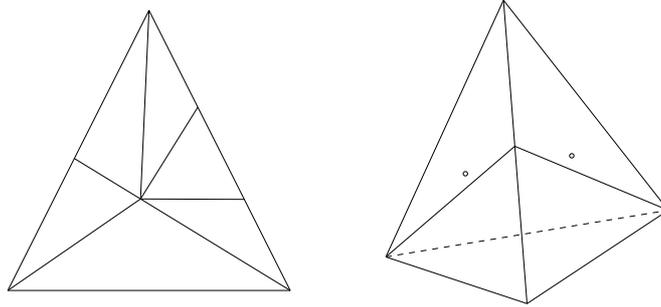


Figura 3.6: Un par de imágenes divididas con aristas de cruce.

La Figura 3.6 presenta un par de imágenes divididas de un 2-simplejo y un 3-simplejo, respectivamente. La primera de ellas tiene una arista de cruce, la arista que conecta las esquinas inferiores. En la segunda, las aristas marcadas con un pequeño círculo son aristas de cruce. A continuación presentamos tres lemas acerca de algunas propiedades que guardan las imágenes divididas sin aristas de cruce. Todos ellos asumen que K^n es una imagen dividida de σ^n bajo ψ , sin aristas de cruce.

Lema 3.1.11 Sean σ_1^m y σ_2^m dos m -caras distintas de σ^n . Considere dos vértices internos u y v de $\psi(\sigma_1^m)$ y $\psi(\sigma_2^m)$, respectivamente. Entonces $u \notin \text{st}(v, \text{bd}(K^n))$.

Demostración: Observe que si $u \in \text{st}(v, \text{bd}(K^n))$, entonces $\{u, v\} \in \text{bd}(K^n)$, lo cual es una contradicción.

□

Lema 3.1.12 Considere un cara propia σ^m de σ^n . Todo m -simplejo de $\psi(\sigma^m)$ contiene por lo menos un vértice interno de $\psi(\sigma^m)$.

Demostración: Observe que $m \leq n - 1$, puesto que σ^m es una cara propia de σ^n . Sea τ un m -simplejo de $\psi(\sigma^m)$. Supongamos, por contradicción, que τ no contiene ningún vértice interno de $\psi(\sigma^m)$. Esto implica que $\text{bd}(\psi(\sigma^m))$ contiene todos los vértices τ . Aún más, por la definición de imagen dividida, Definición 3.1.7 (5), existen $u, v \in \tau$ tales que $u \in \psi(\sigma_1^{m-1})$ y $v \in \psi(\sigma_2^{m-1})$, donde σ_1^{m-1} y σ_2^{m-1} son caras distintas de σ^m . Advierta que $\{u, v\} \in \psi(\sigma^m)$. Por el Lema

3.1.8 (1), $\psi(\sigma^m) \subset bd(\psi(\sigma^n))$, y por el Lema 3.1.9, $\psi(\sigma^n) = K^n$. Es decir, $\{u, v\} \in bd(K^n)$. Una contradicción.

□

Lema 3.1.13 *Considere dos caras propias σ^m y σ^p de σ^n tales que $\sigma^m \subset \sigma^p$. Sea ρ^m un simplejo de $\psi(\sigma^m)$. Considere un simplejo τ^p de $\psi(\sigma^p)$ tal que $\rho^m \subset \tau^p$. Para todo vértice $w \in \tau^p$ tal que $w \notin \rho^m$, existe una cara σ^q de σ^p , $m < q \leq p$, para la cual w es un vértice interno de $\psi(\sigma^q)$.*

Demostración: Primero observe que por la definición de imagen dividida, Definición 3.1.7 (5), τ^p debe existir. Supongamos, por contradicción, que existe un vértice $w \in \tau^p$ tal que el lema no se cumple para él. Es decir, w tiene un carrier de dimensión menor o igual m pero $w \notin \psi(\sigma^m)$. Sea σ'^m una cara de σ^p tal que $w \in \psi(\sigma'^m)$. Sea v cualquier vértice de ρ^m . Advierta que $\{v, w\} \in \psi(\sigma^p)$. El argumento es el mismo que el usado en la demostración del Lema 3.1.12. Por el Lema 3.1.8 (1), $\psi(\sigma^p) \subset bd(\psi(\sigma^n))$. Además, por el Lema 3.1.9, $\psi(\sigma^n) = K^n$. Una contradicción.

□

Imágenes Divididas Cromáticas

Antes de presentar la definición de una imagen dividida cromática y una serie de lemas relativos a su orientación, primero introducimos la definición de coloración de Sperner y el famoso Lema de Sperner, el cual se relaciona con el número de n -simplejos coloreados propiamente en una n -pseudovariedad. Este lema juega un papel preponderante en la demostración de imposibilidad del $(n+1)$ -acuerdo [18, 9], otro problema fundamental en los sistemas distribuidos.

Considere un simplejo σ^n coloreado propiamente con ID^n . Denotamos los colores de una cara σ de σ^n como $ID(\sigma)$. Sea K^n una imagen dividida de σ^n bajo ψ . Decimos que K^n tiene una *coloración de Sperner id* si para todo vértice $v \in K^n$, $id(v) \in ID(carr(v))$. En otras palabras, para una cara σ de σ^n , para todo $u \in \sigma$, $id(\psi(u)) = ID(u)$, y para todo $w \in \psi(\sigma)$, $id(w) \in ID(\sigma)$. Además, decimos que u es una *esquina* de $\psi(\sigma)$. El Teorema 6.1 en [3] es la generalización del Lema de Sperner utilizando el concepto de imágenes divididas. Ahora, reescribimos este lema haciendo uso de nuestra notación.

Lema 3.1.14 (Lema de Sperner) *Sea K^n una imagen dividida de σ^n con una coloración de Sperner. Entonces K^n contiene un número impar de n -simplejos coloreados propiamente bajo id .*

La figura 3.7 muestra una imagen dividida con una coloración de Sperner. Observe que en la imagen dividida, tres 2-simplejos están coloreados propiamente.

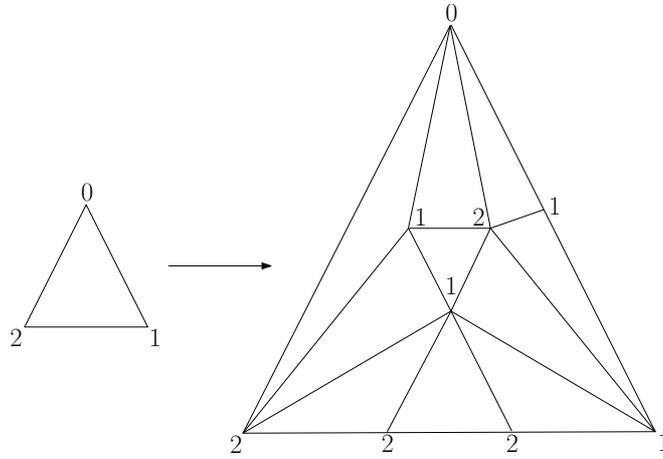


Figura 3.7: El Lema de Sperner.

Sea K^n una imagen dividida de σ^n bajo ψ . Decimos que K^n es una imagen dividida *cromática* si tiene una coloración de Sperner en la cual todo simplejo τ de K^n está coloreado propiamente con los colores de $\text{carr}(\tau)$. En otras palabras, $\text{id}(\tau) = \text{ID}(\text{carr}(\tau))$. Observe que en una imagen dividida cromática, para toda cara σ de σ^n , $\psi(\sigma)$ es una pseudovariiedad cromática coloreada con $\text{ID}(\sigma)$.

Los siguientes dos lemas serán de gran utilidad una vez que definamos, más adelante, las n -esquinas de una imagen dividida. Estos dos asumen que K^n es una imagen dividida cromática y orientada coherentemente de σ^n bajo ψ . Considere la cara σ_i^{n-1} de σ^n . Recuerde que esta es la $(n-1)$ -cara de σ^n sin el vértice coloreado i . Sean ρ_1^{n-1} y ρ_2^{n-1} dos simplejos distintos de $\psi(\sigma_i^{n-1})$ tales que comparten una cara ρ^{n-2} . Por la definición de imagen dividida, Definición 3.1.7 (5), existen los n -simplejos $\tau_1^n, \tau_2^n \in K^n$ tales que $\rho_1^{n-1} \subset \tau_1^n$ y $\rho_2^{n-1} \subset \tau_2^n$.

Lema 3.1.15 *Los simplejos τ_1^n y τ_2^n tienen orientaciones opuestas.*

Demostración: Por la definición de imagen dividida, Definición 3.1.7 (5), K^n debe contener los simplejos $\tau_1^n = \gamma_1^n, \gamma_2^n \dots \gamma_{p-1}^n, \gamma_p^n = \tau_2^n$ tales que todos ellos contienen a ρ^{n-2} , y además γ_k^n y γ_{k+1}^n , $1 \leq k \leq p-1$, comparten una $(n-1)$ -cara ϱ_k^{n-1} , donde $\rho^{n-2} \subset \varrho_k^{n-1}$. Es decir, estos simplejos forman un n -camino de τ_1^n a τ_2^n (ver figura 3.8). De forma intuitiva, éste camino se ve como un abanico que empieza en τ_1^n y termina en τ_2^n . Observe que si éste camino no existiera entonces $\text{bd}(\psi(\sigma^n)) \neq \psi(\text{bd}(\sigma^n))$. Debido a que K^n es cromática, $\text{id}(\tau_1^n) = \text{id}(\tau_2^n) = \text{ID}^n$ e $\text{id}(\rho_1^{n-1}) = \text{id}(\rho_2^{n-1}) = \text{ID}_i^n$. Supongamos, s.p.g., que $\text{id}(\rho^{n-2}) = \text{ID}_{\{i,j\}}^n$. Observe que ϱ_{p-1}^{n-1} únicamente puede estar coloreado con ID_i^n o ID_j^n . Aún más, puesto que $\rho_2^{n-1} \in \text{bd}(K^n)$, tenemos $\text{id}(\varrho_{p-1}^{n-1}) = \text{ID}_j^n$. Es decir, γ_{p-1}^n y $\gamma_p^n = \tau_2^n$ comparten la $(n-1)$ -cara con ID_j^n . Advierta que el subcomplejo que consiste en los n -simplejos $\tau_1^n = \gamma_1^n, \gamma_2^n \dots \gamma_{p-1}^n$ tiene una coloración de Sperner, donde las esquinas son los vértices de ρ_1^{n-1} y el vértice de γ_{p-1}^n coloreado i . Por el Lema de Sperner 3.1.14, sabemos que $p-1$ es impar y, por tanto, p es par. Utilizando

el Lema 3.1.1, podemos inferir que en cualquier orientación coherente τ_1^n y τ_2^n tendrán orientaciones opuestas.

□

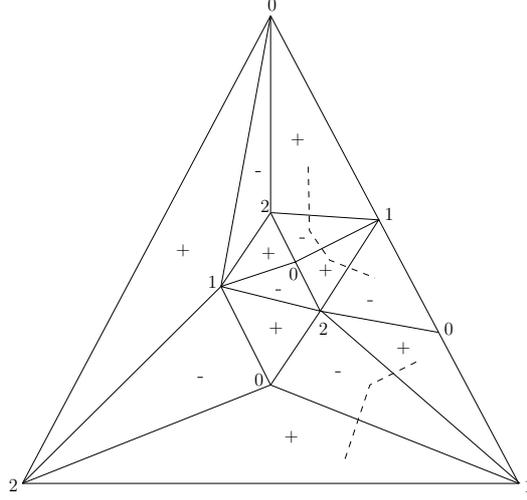


Figura 3.8: Un par de caminos sobre una pseudovariiedad cromática.

Ahora considere dos caras distintas σ_i^{n-1} y σ_j^{n-1} de σ^n . Por la definición de imagen dividida, Definición 3.1.7 (4, 5), existen los simplejos $\rho_1^{n-1} \in \psi(\sigma_i^{n-1})$ y $\rho_2^{n-1} \in \psi(\sigma_j^{n-1})$ tales que comparten un $(n-2)$ -cara ρ^{n-2} , donde $\rho^{n-2} \in \psi(\sigma_i^{n-1} \cap \sigma_j^{n-1})$. Aún más, por la Definición 3.1.7 (5), existen los simplejos τ_1^n , $\tau_2^n \in K^n$ tales que $\rho_1^{n-1} \subset \tau_1^n$ y $\rho_2^{n-1} \subset \tau_2^n$.

Lema 3.1.16 *Los simplejos τ_1^n y τ_2^n tienen la misma orientación.*

Demostración: De forma similar que en la demostración del Lema 3.1.15, por la definición de imagen dividida, Definición 3.1.7 (5), K^n debe contener los simplejos $\tau_1^n = \gamma_1^n, \gamma_2^n \dots \gamma_{p-1}^n, \gamma_p^n = \tau_1^n$ tales que todos ellos contienen a ρ^{n-2} , y además γ_k^n y γ_{k+1}^n , $1 \leq k \leq p-1$, comparten una $(n-1)$ -cara ϱ_k^{n-1} , donde $\rho^{n-2} \subset \varrho_k^{n-1}$ (ver figura 3.8). Observe que $id(\rho_1^{n-1}) = ID_i$ e $id(\rho_2^{n-1}) = ID_j$. Además, la pseudovariiedad formada por el n -camino tiene una coloración de Sperner, donde las esquinas son los vértices de ρ_1^{n-1} y el vértice de ρ_2^{n-1} coloreado i . Por el Lema de Sperner 3.1.14, p es impar. Por lo tanto, por el Lema 3.1.1, τ_1^n y τ_2^n tienen la misma orientación. Advierta que puede ser el caso que $p = 1$ y, por tanto, $\tau_1^n = \tau_2^n$.

□

El siguiente lema muestra una propiedad acerca de la orientación de las imágenes divididas cromáticas que será de utilidad cuando calculemos recursivamente el contenido de una imagen dividida.

Lema 3.1.17 Sea K^n una imagen dividida cromática y orientada coherentemente de σ^n bajo ψ . Entonces K^n induce una orientación coherente a $bd(K^n)$.

Demostración: Primero demostraremos que K^n induce una orientación coherente a $\psi(\sigma_i^{n-1})$, donde σ_i^{n-1} es una cara de σ^n . Por definición, $\psi(\sigma_i^{n-1})$ es una $(n-1)$ -pseudovariiedad cromática y orientable. Además, la orientación de un $(n-1)$ -simplejos de $\psi(\sigma_i^{n-1})$ es la orientación del único n -simplejo en K^n que lo contiene multiplicada por $(-1)^i$. Considere dos simplejos distintos ρ_1^{n-1} y ρ_2^{n-1} de $\psi(\sigma_i^{n-1})$ tales que comparten una $(n-2)$ -cara ρ^{n-2} . Sean τ_1^n y τ_2^n los simplejos en K^n tales que $\rho_1^{n-1} \subset \tau_1^n$ y $\rho_2^{n-1} \subset \tau_2^n$. Observe que ρ_1^{n-1} induce a ρ^{n-2} su orientación multiplicada por $(-1)^j$, donde j es el color del único vértice que pertenece a ρ_1^{n-1} y no a ρ^{n-2} . De forma similar ocurre con ρ_2^{n-1} y ρ^{n-2} . Por el lema 3.1.15, τ_1^n y τ_2^n tienen orientaciones opuestas. Por lo tanto, ρ_1^{n-1} y ρ_2^{n-1} tienen orientaciones opuestas y ρ^{n-2} adquiere orientaciones opuestas de ellos.

Ahora demostraremos que K^n induce una orientación coherente a $\psi(\sigma_i^{n-1}) \cup \psi(\sigma_j^{n-1})$, donde σ_i^{n-1} y σ_j^{n-1} son caras distintas de σ^n . Primero observe que el Lema 3.1.1 no se cumple para $\psi(\sigma_i^{n-1}) \cup \psi(\sigma_j^{n-1})$, ya que esta es una $(n-1)$ -pseudovariiedad coloreada con $n+1$ colores. Considere dos simplejos ρ_1^{n-1} y ρ_2^{n-1} de $\psi(\sigma_i^{n-1})$ y $\psi(\sigma_j^{n-1})$, respectivamente, tales que comparten una $(n-2)$ -cara ρ^{n-2} , donde $\rho^{n-2} \in \psi(\sigma_i^{n-1} \cap \sigma_j^{n-1})$. Sean τ_1^n y τ_2^n los simplejos de K^n tales que $\rho_1^{n-1} \subset \tau_1^n$ y $\rho_2^{n-1} \subset \tau_2^n$. Observe que $id(\tau_1^n) = ID_i^n$, $id(\tau_2^n) = ID_j^n$ e $id(\rho^{n-2}) = ID_{\{i,j\}}^n$. Además, la orientación de ρ_1^{n-1} es la orientación de τ_1^n multiplicada por $(-1)^i$ y la orientación de ρ_2^{n-1} es la orientación de τ_2^n multiplicada por $(-1)^j$. Por el Lema 3.1.16, τ_1^n y τ_2^n tiene la misma orientación. Supongamos, s.p.g, que τ_1^n y τ_2^n tiene orientación $+1$ e $i < j$. Denotamos los vértices de ρ_1^{n-1} como $v_0, v_1 \dots v_{n-1}$, de forma tal que $id(v_0) = 0, id(v_1) = 1 \dots id(v_{i-1}) = i-1, id(v_i) = i+1 \dots id(v_{n-1}) = n$. Observe que v_{j-1} es el vértice que pertenece a ρ_1^{n-1} pero no a ρ^{n-2} . Por lo tanto, la orientación que induce ρ_1^{n-1} a ρ^{n-2} es $(-1)^i(-1)^{j-1}$. De forma similar, denotamos los vértices de ρ_2^{n-1} como $v_0, v_1 \dots v_{n-1}$, de forma tal que $id(v_0) = 0, id(v_1) = 1 \dots id(v_{j-1}) = j-1, id(v_j) = j+1 \dots id(v_{n-1}) = n$. Advierta que v_i es el vértice que pertenece a ρ_2^{n-1} pero no a ρ^{n-2} . Por lo tanto, la orientación que induce ρ_2^{n-1} a ρ^{n-2} es $(-1)^j(-1)^i$. Esto completa la demostración. □

Imágenes Divididas Simétricas

En esta sección presentamos el concepto de simetría en la imágenes divididas. Intuitivamente, en una imagen dividida simétrica cualquier par de imágenes divididas de caras de la misma dimensión tienen la misma subdivisión. También presentamos un importante lema que muestra la relación que guardan las orientaciones de los simplejos en la frontera de una imagen dividida simétrica.

Considere una imagen dividida cromática K^n de σ^n bajo ψ . Decimos que K^n tiene *estructura simétrica* si para todo par de caras τ^m y ρ^m de σ^n , existe una biyección simplicial f de $\psi(\tau^m)$ a $\psi(\rho^m)$ que *preserva ids*. Por esto entendemos

que para todo par $u, v \in \psi(\tau^m)$, si $id(u) = id(v)$ entonces $id(f(u)) = id(f(v))$. Observe que debido a que K^n es cromática, la inversa de f , f^{-1} , también es una biyección simplicial de $\psi(\rho^m)$ a $\psi(\tau^m)$ que preserva *ids*. Asimismo, K^n tiene estructura binaria simétrica si tiene una coloración binaria b tal que para todo $v \in bd(K^n)$, $b(v) = b(f(v))$. Es decir, b preserva colores. Advierta que pueden existir varias biyecciones para las cuales la condición de simetría se cumple, y, por tanto, la simetría de b depende de la elección de f . En lo subsecuente, diremos que dos vértices $u \in \psi(\tau^m)$ y $v \in \psi(\rho^m)$ son *isomorfos con respecto a f* si $f(u) = v$. De forma similar definimos los simplejos isomorfos. Por lo tanto, un vértice o simplejo de $\psi(\tau^m)$ puede ser isomorfo a varios vértices o simplejos de $\psi(\rho^m)$, dependiendo de la f que se elija.

Lema 3.1.18 *Sea K^n una imagen dividida cromática de σ^n bajo ψ , con estructura simétrica. Considere las caras τ^m y ρ^m de σ^n . Sea f una biyección simplicial de $\psi(\tau^m)$ a $\psi(\rho^m)$ que preserva *ids*. Para toda cara τ^{m-1} de τ^m existe una cara ρ^{m-1} de ρ^m tal que $f|_{\psi(\tau^{m-1})}$ es una biyección simplicial de $\psi(\tau^{m-1})$ a $\psi(\rho^{m-1})$ que preserva *ids*.*

Demostración: Primero observe que $f|_{\psi(\tau^{m-1})}$ preserva *ids*. Además, todo simplejo de $\psi(\tau^m)$ es mapeado a un simplejo de $\psi(\rho^m)$, ya que f es simplicial. En particular, todo simplejo de $bd(\psi(\tau^m))$ es mapeado a un simplejo de $bd(\psi(\rho^m))$. Aún más, ya que f es una biyección, los simplejos isomorfos con respecto a f tienen la misma dimensión. Ahora supongamos, por contradicción, que ρ^{m-1} no existe. Esto es, $f|_{\psi(\tau^{m-1})}$ va de $\psi(\tau^{m-1})$ a L^{m-1} , donde L^{m-1} es un subsimplejo de $\psi(\rho_1) \cup \psi(\rho_2)$ para algún par de $(m-1)$ -caras distintas ρ_1 y ρ_2 de ρ^m . En otras palabras, algunos simplejos de $\psi(\tau^{m-1})$ son mapeados a $\psi(\rho_1)$ y los demás a $\psi(\rho_2)$. Sean γ_1 y γ_2 simplejos de $\psi(\tau^{m-1})$ tales que $f(\gamma_1) \in \psi(\rho_1)$ y $f(\gamma_2) \in \psi(\rho_2)$. Tenemos que $id(\gamma_1) = id(\gamma_2) = ID(\tau^{m-1})$, $id(\psi(\gamma_1)) = ID(\rho_1)$ e $id(\psi(\gamma_2)) = ID(\rho_2)$. No obstante, $ID(\rho_1) \neq ID(\rho_2)$. Por lo tanto, existen los vértices $u \in \gamma_1$ y $v \in \gamma_2$ tales que $id(u) = id(v)$ e $id(f(u)) \neq id(f(v))$. Una contradicción. □

Ahora definimos un caso especial de mapeos bajo los cuales una imagen dividida puede tener estructura simétrica. Sea K^n una imagen dividida cromática de σ^n bajo ψ . Considere dos caras τ^m y ρ^m de σ^n . Una biyección simplicial f de $\psi(\tau^m)$ a $\psi(\rho^m)$ preserva orden si para todo $v \in \psi(\tau^m)$ se cumple $rank(id(v), ID(\tau^m)) = rank(id(f(v)), ID(\rho^m))$. Esto es, los vértices de $\psi(\tau^m)$ se mapean a los vértices de $\psi(\rho^m)$ con el mismo *rank*. Observe que si f preserva orden entonces también preserva *ids*. Todavía más, ya que σ^n está coloreado propiamente, f es única. En lo subsecuente, únicamente consideraremos imágenes divididas que tienen estructuras simétrica bajo mapeos que preservan orden.

Sean τ^m y ρ^m dos caras de σ^n y f una biyección simplicial de $\psi(\tau^m)$ a $\psi(\rho^m)$ que preserva orden. Por el Lema 3.1.18, para toda cara τ^{m-1} de τ^m existe la cara ρ^{m-1} de ρ^m tal que $f|_{\psi(\tau^{m-1})}$ es una biyección simplicial de $\psi(\tau^{m-1})$ a $\psi(\rho^{m-1})$ que preserva orden.

Afirmación 3.1.19 Sea h una biyección simplicial de $\psi(\tau^m)$ a $\psi(\rho^m)$ que preserva orden. Entonces $h = f|_{\psi(\tau^{m-1})}$ y $h^{-1} = f|_{\psi(\tau^{m-1})}^{-1}$.

Es decir, un simplejo de $\psi(\tau^{m-1})$ es isomorfo a un único simplejo de $\psi(\rho^{m-1})$. De aquí en adelante, cuando mencionemos que un par de vértices $u \in \psi(\tau^m)$ y $v \in \psi(\rho^m)$ son isomorfos, nos referiremos a que lo son con respecto a la biyección simplicial $f : \psi(\tau^m) \rightarrow \psi(\rho^m)$ que preserva orden.

A continuación definimos las n -esquinas de una imagen dividida. Informalmente, estas son los n -simplejos que tienen una cara en la frontera para cualquier dimensión. Como veremos más adelante, estos n -simplejos tienen la misma orientación cuando la imagen dividida es cromática y orientable.

Definición 3.1.20 Considere una imagen dividida K^n de σ^n bajo ψ . El conjunto de n -esquinas de K^n es

$$n\text{-esquinas}(K^n) = \{ \tau^n \in K^n \mid \forall 0 \leq m \leq n, \exists \sigma^m \subseteq \tau^n \text{ tal que } \rho^m \subseteq \tau^n \text{ para algún } \rho^m \in \psi(\sigma^m) \}$$

No es difícil ver que $n\text{-esquinas}(K^n) \neq \emptyset$. En la Figura 3.9 se puede ver una imagen dividida con sus n -esquinas marcadas con un pequeño círculo.

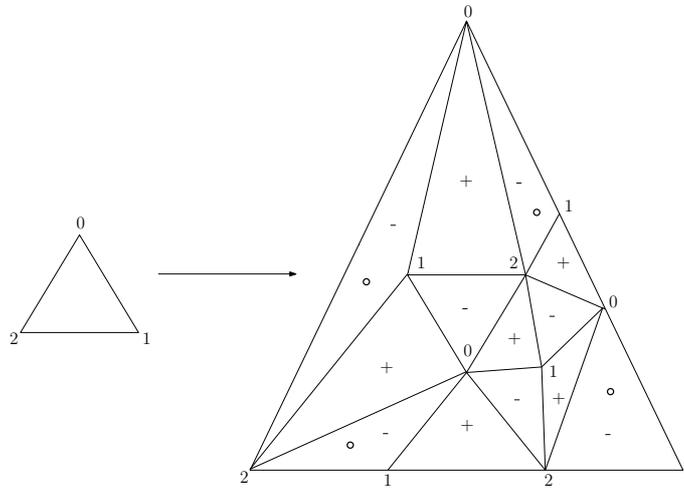


Figura 3.9: Las 2-esquinas de una imagen dividida de un 2-simplejo

El siguiente lema presenta una propiedad importante de las orientaciones de las n -esquinas de una imagen dividida.

Lema 3.1.21 Sea K^n imagen dividida cromática, conexa y orientada coherentemente de σ^n bajo ψ . Todo par de simplejos σ_1^n y σ_2^n de n -esquinas(K^n) tienen la misma orientación.

Demostración: Considere las caras $\sigma_2^1, \sigma_3^2 \dots \sigma_n^{n-1}$ de σ^n tales que $\sigma_2^1 \subset \sigma_3^2 \subset \dots \subset \sigma_n^{n-1}$. Esto es, $ID(\sigma_2^1) = \{0, 1\}$, $ID(\sigma_3^2) = \{0, 1, 2\}$, y así sucesivamente. Para $1 \leq i \leq n-2$, asumimos que $\psi(\sigma_{i+1}^i)$ tiene la orientación inducida por σ_{i+2}^{i+1} . Además, σ_n^{n-1} tiene la orientación inducida por K^n . Por el Lema 3.1.17, todas estas son orientaciones coherentes. Procederemos por inducción en n . La base de la inducción es para σ_2^1 . Ya que K^n es una imagen dividida conexa, $\psi(\sigma_2^1)$ es un 1-camino que además tiene una coloración de Sperner. Por el Lema de Sperner 3.1.14, $\psi(\sigma_2^1)$ tiene un número impar de 1-simplejos. Por el Lema 3.1.1, los simplejos en 1 -esquinas($\psi(\sigma_2^1)$) tienen la misma orientación. Note que en éste caso 1 -esquinas($\psi(\sigma_2^1)$) contiene únicamente uno o dos simplejos. Supongamos que el lema se cumple para $m-1$. Demostraremos que también se cumple para m . De la definición de n -esquinas, Definición 3.1.20, podemos ver que todo simplejo de $(m-1)$ -esquinas($\psi(\sigma_m^{m-1})$) es cara de un simplejo de m -esquinas($\psi(\sigma_{m+1}^m)$). Además, todo simplejo de m -esquinas($\psi(\sigma_{m+1}^m)$) contiene un simplejo de $(m-1)$ -esquinas($\psi(\sigma_m^{m-1})$).

Primero considere dos simplejos ρ_1^{m-1} y ρ_2^{m-1} de $(m-1)$ -esquinas($\psi(\sigma_m^{m-1})$). Sean τ_1^m y τ_2^m los simplejos de m -esquinas($\psi(\sigma_{m+1}^m)$) tales que $\rho_1^{m-1} \subset \tau_1^m$ y $\rho_2^{m-1} \subset \tau_2^m$. Recuerde que los simplejos de $\psi(\sigma_{m+1}^m)$ y $\psi(\sigma_m^{m-1})$ están coloreados propiamente con $ID(\sigma_{m+1}^m)$ e $ID(\sigma_m^{m-1})$ bajo id , respectivamente. Observe que τ_1^m induce su orientación multiplicada por $(-1)^m$ a ρ_1^{m-1} . De forma similar ocurre con τ_2^m y ρ_2^{m-1} . Por hipótesis inductiva, tenemos que los simplejos de $(m-1)$ -esquinas($\psi(\sigma_m^{m-1})$) tienen la misma orientación y, por tanto, τ_1^m y τ_2^m tienen la misma orientación.

Considere una cara σ^{m-1} de σ_{m+1}^m tal que $\sigma^{m-1} \neq \sigma_m^{m-1}$. Ahora demostraremos que las m -esquinas de $\psi(\sigma_{m+1}^m)$ que contienen una m -esquina de $\psi(\sigma_m^{m-1})$ o $\psi(\sigma^{m-1})$ tienen la misma orientación. Considere dos simplejos $\rho_1^{m-1} \in (m-1)$ -esquinas($\psi(\sigma_m^{m-1})$) y $\rho_2^{m-1} \in (m-1)$ -esquinas($\psi(\sigma^{m-1})$). Sean τ_1^m y τ_2^m los simplejos de m -esquinas($\psi(\sigma_{m+1}^m)$) tales que $\rho_1^{m-1} \subset \tau_1^m$ y $\rho_2^{m-1} \subset \tau_2^m$. Note que σ_m^{m-1} y σ^{m-1} comparten una $(m-2)$ -cara σ^{m-2} . Considere el simplejo $\tau^{m-2} \in \psi(\sigma^{m-2})$ tal que $\tau^{m-2} \subset \rho_1^{m-1}$. Sea ρ_3^{m-1} el simplejo de $(m-1)$ -esquinas($\psi(\sigma^{m-1})$) tal que $\tau^{m-2} \subset \rho_3^{m-1}$. Ahora considere el simplejo τ_3^m de m -esquinas($\psi(\sigma_{m+1}^m)$) tal que $\rho_3^{m-1} \subset \tau_3^m$. Por el Lema 3.1.16, sabemos que τ_1^m y τ_3^m tienen la misma orientación. Por el caso anterior y puesto que $\psi(\sigma^{m-1})$ tiene la orientación inducida por $\psi(\sigma_{m+1}^m)$, tenemos que τ_2^m y τ_3^m tienen la misma orientación. Esto completa la demostración. \square

En la Figura 3.9 se puede ver una imagen dividida cromática, conexa y orientada coherentemente con sus 2-esquinas marcadas con un pequeño círculo. Observe que estas últimas tienen orientación -1. El siguiente lema muestra la relación que guardan las orientaciones de los simplejos isomorfos en la frontera de una imagen dividida cromática con estructura simétrica. Este lema será de gran importancia en el capítulo 4.

Lema 3.1.22 Sea K^n una imagen dividida cromática, conexa y orientada coherentemente de σ^n bajo ψ , con estructura simétrica. Considere las caras σ_i^{n-1}

y σ_j^{n-1} de σ^n . Sean ρ_1^{n-1} y ρ_2^{n-1} simplejos isomorfos de $\psi(\sigma_i^{n-1})$ y $\psi(\sigma_j^{n-1})$, respectivamente. Los n -simplejos de K^n que contienen a ρ_1^{n-1} y ρ_2^{n-1} , respectivamente, tiene la misma orientación.

Demostración: Sean $\tau_1^n, \tau_2^n \in K^n$ tales que $\rho_1^{n-1} \subset \tau_1^n$ y $\rho_2^{n-1} \subset \tau_2^n$. Observe que la orientación de ρ_1^{n-1} es la orientación de τ_1^n multiplicada por $(-1)^i$, mientras que la orientación de ρ_2^{n-1} es la orientación de τ_2^n multiplicada por $(-1)^j$. Por lo tanto, nos basta demostrar que las orientaciones de ρ_1^{n-1} y ρ_2^{n-1} multiplicadas por $(-1)^i$ y $(-1)^j$, respectivamente, son iguales.

Por la definición de n -esquinas, Definición 3.1.20, tenemos que todo simplejo de $(n-1)$ -esquinas($\psi(\sigma_i^{n-1})$) o $(n-1)$ -esquinas($\psi(\sigma_j^{n-1})$) es cara de un simplejo de n -esquinas($\psi(\sigma^n)$). Considere dos simplejos τ_1^{n-1} y τ_2^{n-1} de $(n-1)$ -esquinas($\psi(\sigma_i^{n-1})$) y $(n-1)$ -esquinas($\psi(\sigma_j^{n-1})$), respectivamente. Por el Lema 3.1.21, las orientaciones de τ_1^{n-1} y τ_2^{n-1} multiplicadas por $(-1)^i$ y $(-1)^j$, respectivamente, son iguales. Utilizando el Lema 3.1.1 podemos deducir que $\psi(\sigma_i^{n-1})$ y $\psi(\sigma_j^{n-1})$ tienen únicamente dos posibles orientaciones, respectivamente (recuerde que K^n es conexa). Por lo tanto, si una biyección simplicial f de $\psi(\sigma_i^{n-1})$ a $\psi(\sigma_j^{n-1})$ que preserva *ids* mapea un simplejo de $(n-1)$ -esquinas($\psi(\sigma_i^{n-1})$) a otro de $(n-1)$ -esquinas($\psi(\sigma_j^{n-1})$), entonces todo $(n-1)$ -simplejo de $\psi(\sigma_i^{n-1})$ es isomorfo con respecto a f a otro de $\psi(\sigma_j^{n-1})$ con la misma orientación, multiplicándolas por $(-1)^i$ y $(-1)^j$, respectivamente (recuerde que las n -esquinas de K^n tienen la misma orientación).

Lo único que nos falta demostrar es que f mapea un simplejo en $(n-1)$ -esquinas($\psi(\sigma_i^{n-1})$) a otro en $(n-1)$ -esquinas($\psi(\sigma_j^{n-1})$). Procederemos por inducción en n . Para la base de la inducción, considere dos caras τ_1^1 y τ_2^1 de σ^n . Es claro que un simplejo en 1 -esquinas($\psi(\tau_1^1)$) forzosamente debe ser mapeado a otro en 1 -esquinas($\psi(\tau_2^1)$). Supongamos que la condición se cumple para $m-1$. Demostraremos que se sigue cumpliendo para m . Sean σ_1^m, σ_2^m y σ_1^{m-1} caras de σ^n tales que $\sigma_1^{m-1} \subset \sigma_1^m$. Por el Lema 3.1.18, sabemos que para σ_1^{m-1} existe $\sigma_2^{m-1} \subset \sigma_2^m$ tal que los vértices de $\sigma(\rho_1^{m-1})$ son mapeados a los vértices de $\psi(\sigma_2^{m-1})$. Por la definición de n -esquinas, Definición 3.1.20, todo simplejo en $(m-1)$ -esquinas($\psi(\sigma_1^{m-1})$) es cara de un simplejo en m -esquinas($\psi(\sigma_1^m)$). De forma similar ocurre con los simplejos en $(m-1)$ -esquinas($\psi(\sigma_2^{m-1})$) y m -esquinas($\psi(\sigma_2^m)$). Aún más, por hipótesis de inducción, un simplejo de $(m-1)$ -esquinas($\psi(\sigma_1^{m-1})$) es mapeado a otro en $(m-1)$ -esquinas($\psi(\sigma_2^{m-1})$). Observe que la condición se cumple porque f es simplicial. Esto completa la demostración.

□

Advierta que en la demostración del lema anterior no es necesario asumir que f es una biyección simplicial que preserva orden.

3.2. Modelado de Tareas y Algoritmos

Ahora es tiempo de describir la forma en que se relacionan los sistemas distribuidos y la topología. De esta relación surge una condición de solubilidad de una tarea por un algoritmo libre espera en el modelo asíncrono de memoria compartida. También presentamos el tipo de imágenes divididas que induce un algoritmo de renombrado en ese modelo, las que serán nuestro objeto de estudio en el capítulo 4.

3.2.1. Modelado de Tareas

Sea Δ una tarea (ver sección 2.1.3) con los dominios de valores de entrada y salida I y O , respectivamente. A continuación mostramos como hacer una definición formal de Δ haciendo uso de dos complejos que representan a I y O y un mapeo simplicial entre estos dos.

Definimos al complejo de entrada I^n como el complejo que contiene un n -simplejo σ , junto con todas sus caras, por cada vector de entradas validas sobre I , de forma tal que σ está coloreado propiamente con ID^n y, además, el vértice de σ coloreado i tiene asociado el valor de entrada del procesador p_i . El complejo de salida O^n se define de forma similar. En este sentido, un vértice (i, val_1) en I^n indica que el procesador p_i tiene el valor de entrada val_1 , mientras que un vértice (j, val_2) en O^n indica que el procesador p_j tiene el valor de salida val_2 . Observe que los complejos I^n y O^n tienen una coloración propia en los identificadores de los procesadores, pero no necesariamente así en los valores de entrada y salida, respectivamente. Además, por construcción, son completos para dimensión n .

Modelamos la tarea Δ como una tripleta $\langle I^n, O^n, \Delta' \rangle$, donde I^n y O^n son los complejos de entrada y salida, respectivamente, y Δ' es un mapeo que asigna a todo n -simplejo de I^n un subconjunto no vacío de n -simplejos de O^n , el cual es congruente con Δ . En este sentido, $\Delta'(\sigma^n)$ son las salidas permitidas por Δ para las entradas representadas por el simplejo de entrada σ^n . En otras palabras, si $\tau^n \in \Delta'(\sigma^n)$, entonces los procesadores pueden tener las salidas en τ^n cuando el sistema comienza con las entradas en σ^n .

Por ejemplo, la Figura 3.10 muestra la forma en que se modela la tarea del consenso binario para dos procesadores, P y Q (ver sección 2.1.3). El complejo de entrada O^1 contiene los simplejos de todas las entradas validas a la tarea, mientras que el complejo de salida I^1 contiene los simplejos de todas las salidas validas. El mapeo Δ asocia cada simplejo de entrada a las posibles salidas que pueden tener los procesadores para esa entrada en específico.

Para el caso del renombrado tenemos lo siguiente. El complejo de entrada I^n es el complejo que contiene todos los n -simplejos, junto con todas sus caras, formados por los subconjuntos de tamaño $n + 1$ del dominio de nombres originales I . De forma similar se define el complejo de salida O^n sobre el dominio de nombres nuevos O . El mapeo Δ mapea todo n -simplejo de I^n a todos los n -simplejos de O^n . En la Figura 1.3 se puede ver el complejo de salida

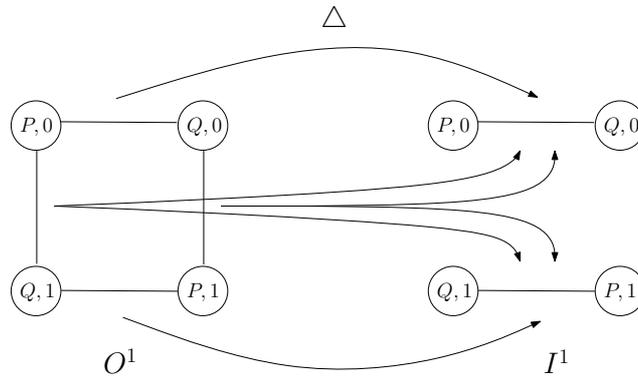


Figura 3.10: Complejos del consenso binario.

para tres procesadores, P , Q y R , con el dominio de nombres nuevos $\{0, 1, 2\}$. Otro ejemplo se muestra en la Figura 1.4, el cual ocupa el dominio de nombres nuevos $\{0, 1, 2, 3\}$. Observe que si los dominios de nombres originales y nuevos son $\{0, 1, 2, 3\}$ y $\{0, 1, 2\}$, respectivamente, entonces el complejo de entrada es el que se muestra en la Figura 1.4 y el de salida el que se ve en la Figura 1.3.

3.2.2. Modelado de Algoritmos

Así como modelamos tareas, también podemos modelar algoritmos. Para este caso, representamos las ejecuciones de un algoritmo en las que los procesadores han decidido.

El complejo A^n de un algoritmo A es el complejo que consiste en un n -simplejo, junto con todas sus caras, por cada posible ejecución del algoritmo en la que todos los procesadores del sistema han decidido (que están en un estado final). Esto es, cada vértice del simplejo está coloreado con el identificador, la entrada y la vista del procesador en la ejecución. De forma similar que cuando se modelan tareas, la coloración de identificadores debe ser propia, mientras que las otras dos pueden no serlo. Por como fue construido, si dos n -simplejos σ_1 y σ_2 de A^n comparten una cara τ , entonces los procesadores en τ no pueden distinguir entre las ejecuciones que representan σ_1 y σ_2 (ver sección 2.1.1).

Sea σ un n -simplejo de entrada. Denotamos como $A^n(\sigma)$ al subcomplejo de A^n que contiene los n -simplejos, junto con todas sus caras, que corresponden a las ejecuciones de A en las que los procesadores comienzan con las entradas en σ . Escrito de otra forma, un simplejo τ pertenece a $A^n(\sigma)$ si y solo si existe una ejecución α de A , tal que las vistas de los procesadores son las mismas que en τ y los procesadores comienzan con las entradas en σ . En [17] se muestran diversos ejemplos de complejos de algoritmos en modelos tanto síncrono como asíncrono.

Observe que al definir el complejo de un algoritmo, en ningún momento hicimos alusión a las salidas que deciden los procesadores. Esto es así, porque el algoritmo A implica el mapeo de decisión $\delta_A : A^n \rightarrow O^n$ que especifica la

salida de un procesador cuando está en un estado final específico. Aún más, debido a que estamos asumiendo que los algoritmos únicamente dependen de sus entradas (ver sección 2.1.1), si σ y σ' son dos n -simplejos de entrada con los mismos valores, es decir, difieren únicamente por una permutación de los identificadores, entonces $A^n(\sigma)$ puede ser obtenido a partir de $A^n(\sigma')$ aplicando la misma permutación. Esto es, δ_A es *anónimo* y, por tanto, $\delta_A(A^n(\sigma))$ determina a $\delta_A(A^n(\sigma'))$.

3.2.3. Condición de Solubilidad

Con lo expuesto en las dos secciones anteriores, podemos hacer una descripción topológica precisa de la condición que ha de cumplir un algoritmo que dice solucionar una tarea en específico.

Afirmación 3.2.1 ([3, Afirmación 4.5]) *Decimos que el algoritmo A solución la tarea $\langle I^n, O^n, \Delta \rangle$ si y solo si para todo n -simplejo de entrada $\sigma \in I^n$ se cumple $\delta_A(A^n(\sigma)) \subseteq \Delta(\sigma)$. Si es el caso, decimos que δ_A es congruente con Δ .*

La afirmación anterior es una vía formal de decir lo siguiente: si una algoritmo soluciona una tarea, entonces en cualquier ejecución los procesadores deben decidir salidas que sean permitidas por la especificación de la tarea. Hasta este punto parecería ser simplemente una forma muy engorrosa de expresar algo que es bastante intuitivo, sin embargo, lo que hemos logrado con esto es pasar a un campo de las matemáticas, la topología, en la cual podremos obtener distintos resultados acerca de los sistemas distribuidos a través de las diferentes técnicas que se conocen en ella.

Ahora podemos describir cual es la estrategia que seguiremos cuando el objetivo sea demostrar la imposibilidad de alguna tarea. A saber, para demostrar que no existe un algoritmo que solucione una tarea dada, nos basta con probar que no existe el mapeo del que se habla en la Afirmación 3.2.1. En otras palabras, lo que tenemos que hacer es demostrar que el complejo que genera un algoritmo en cierto modelo, tiene ciertas características topológicas que hacen imposible que exista un mapeo de este hacia el complejo de salida y que además, sea congruente con la tarea.

3.2.4. El Complejo de las Ejecuciones *Immediate Snapshot*

Como se explicó en la sección 2.1.4, las ejecuciones IS son un subconjunto del conjunto de todas las ejecuciones posibles en el modelo de memoria compartida. Aún más, restringir un algoritmo a este tipo de ejecuciones no limita en nada su poder de cómputo. El complejo de las ejecuciones IS, E^n , de un algoritmo A , es el subcomplejo de A^n que contiene los n -simplejos correspondientes a las ejecuciones IS. De esta forma, denotamos como $E^n(\sigma^n)$ a las ejecuciones IS que se obtienen cuando los procesadores comienzan con las entradas en $\sigma^n \in I^n$. La

propiedad fundamental de E^n , es que es una imagen dividida cromática, orientable y conexa del complejo de entrada I^n [3]. Dado que las ejecuciones IS son igualmente poderosas que el modelo general de memoria compartida, tenemos lo siguiente.

Teorema 3.2.2 ([3, Teorema 5.14]) *Sea $\langle I^n, O^n, \Delta \rangle$ una tarea. Si existe un algoritmo libre de espera que soluciona esta tarea, entonces existe una imagen dividida cromática, orientable y conexa de I^n , y además existe un mapeo simplicial anónimo que preserva colores (en identificadores de procesadores) que va de la imagen dividida a O^n y que es congruente con Δ .*

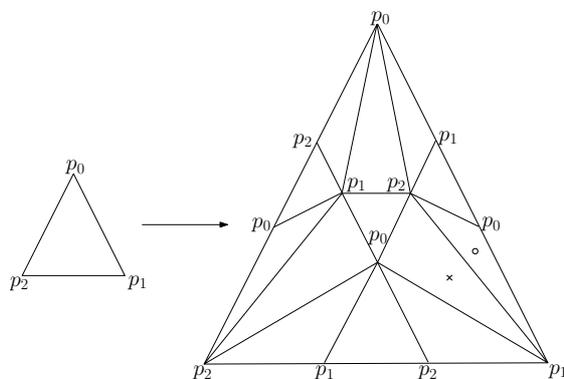


Figura 3.11: Complejo de ejecuciones IS en el que los procesadores están activos una vez.

En la Figura 3.11 se puede ver el complejo de ejecuciones IS para un algoritmo en el que los procesadores están activos exactamente una vez. El triángulo de la izquierda representa una entrada al algoritmo y la imagen dividida las ejecuciones IS posibles. Por ejemplo, las ejecuciones α_1 y α_2 de la Figura 2.3 se representan por los triángulos marcados con un círculo y una cruz, respectivamente. Además, las aristas en la frontera representan las ejecuciones en las que solo dos procesadores están activos. Esto es, las tres aristas del segmento de frontera que va de la esquina p_0 a p_1 , son las ejecuciones en las que solo estos dos procesadores han estado activos. La arista de en medio es la ejecución en la que p_0 y p_1 están activos al mismo tiempo, mientras que las otras dos son las ejecuciones en las que primero está activo uno y después el otro. De forma similar, las esquinas son las ejecuciones en las que un procesador sólo se ve a sí mismo.

De acuerdo a lo anterior, el complejo del algoritmo *AlgRenombrado1* presentado en la sección 2.2.3, con las entradas a , b y c , es el que se muestra en la Figura 3.12. En el complejo se muestra la salida de los procesadores en lugar de su estado. Advierta que las decisiones de los procesadores da un mapeo simplicial anónimo que va de la imagen dividida al complejo de salida y, además, es congruente con la tarea, tal como lo afirma el Teorema 3.2.2. Aún podemos

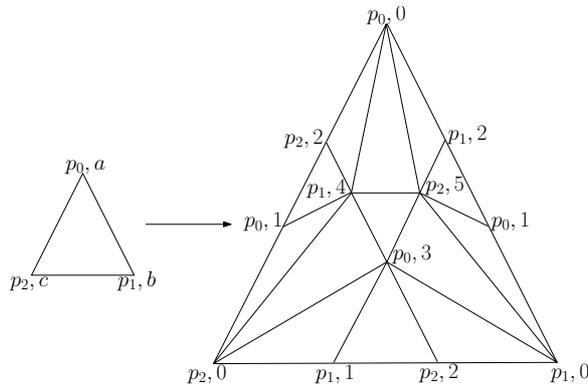


Figura 3.12: Complejo del algoritmo *AlgRenombrado1*.

hacer una observación más. El patrón de salidas que se lee en el segmento de frontera que va de la esquina p_0 a p_1 , es el mismo que se lee en el segmento de frontera que va de la esquina p_1 a p_2 . Además, la entrada de p_0 es menor que la de p_1 , que a su vez es menor que la de p_2 . Esto no es casualidad, ya que las únicas operaciones que realiza el algoritmo entre las entradas son comparaciones y, por tanto, un procesador ni siquiera puede saber cuales son los valores que está comparando. Por ende, el algoritmo decide las mismas salidas en ejecuciones en las cuales las entradas guardan el mismo orden. Esta acotación será de gran importancia en la siguiente sección.

La principal implicación del Teorema 3.2.2, es que para probar que no existe un algoritmo libre de espera que solucione cierta tarea, basta demostrar que no existe un mapeo simplicial congruente con la tarea, que va de una imagen dividida como la que pide el teorema, al complejo de salida. Aún más, como se muestra en [7, 8, 18], la dirección opuesta también es cierta. A saber, si existe una imagen dividida como la que pide el teorema y, además, existe un mapeo simplicial congruente con la tarea que va de la imagen dividida al complejo de salida, entonces existe un algoritmo libre de espera que soluciona la tarea. Utilizaremos fuertemente estas ideas en el capítulo 5 para obtener una demostración de imposibilidad del k -renombrado, $k < 2n$.

3.2.5. El Renombrado y las Imágenes Divididas

Con lo visto hasta ahora, podemos hacer una descripción precisa del tipo de imágenes divididas que implican los algoritmos de renombrado que utilizan ejecuciones IS. Antes de esto, presentamos el siguiente lema, el cual se obtiene directamente del Lema 8.7 en [17].

Lema 3.2.3 *Existe un algoritmo libre de espera que soluciona el k -renombrado si y solo si existe un algoritmo libre de espera basado en comparaciones que soluciona el k -renombrado.*

Supongamos que existe un algoritmo A que soluciona el k -renombrado. Asumiremos que el dominio de nombres nuevos I es finito y, por tanto, I^n es finito. Esto implica que en I^n existe un simplejo de entrada para el cual los procesadores deben realizar un número mayor o igual de pasos para decidir su salida que con todos los demás simplejos de entrada. En otras palabras, este es el peor caso, en cuanto a tiempo, de A . Por lo tanto, podemos asumir que en todas las ejecuciones de A , los procesadores realizan este número de pasos. Por ejemplo, aunque un procesador ya haya decidido, simplemente sigue leyendo y escribiendo en la memoria hasta completar los pasos necesarios. Aún más, por el Lema 3.2.3, podemos asumir que A es basado en comparaciones.

De acuerdo al Teorema 3.2.2, el algoritmo A implica que existe una imagen dividida L^n cromática, orientable y conexa del complejo de entrada I^n . Como ya vimos en la sección anterior, esta imagen dividida representa el complejo de ejecuciones IS. Sea ϕ el mapeo bajo el cual L^n es imagen dividida de I^n . Considere un simplejo de entrada $\sigma^n \in I^n$. Por el Lema 3.1.8 (3), $\phi(\sigma^n)$ es una imagen dividida de σ^n bajo $\phi|_{M(\sigma^n)}$. Todavía mas, $\phi(\sigma^n)$ es cromática, orientable y conexa. Denotaremos a $\phi(\sigma^n)$ como K^n y a $\phi|_{M(\sigma^n)}$ como ψ .

Considere dos caras σ_1^m y σ_2^m de σ^n . Recuerde que $\psi(\sigma_1^m)$ representa las ejecuciones en las cuales únicamente los procesadores en σ_1^m están activos. De igual forma sucede con σ_2^m . Puesto que todos los procesadores ejecutan el mismo número de pasos, $\psi(\sigma_1^m)$ y $\psi(\sigma_2^m)$ tienen la misma subdivisión y, por tanto, existe una biyección simplicial entre ellos. Ahora considere el mapeo de decisión δ_A inducido por A . Agregamos a K^n la coloración binaria b que asocia a cada vértice la paridad del nombre nuevo decidido por δ_A . Ya que el algoritmo A es basado en comparaciones y anónimo, la coloración b preserva colores con respecto a la biyección simplicial con preservación de orden $f : \psi(\sigma_1^m) \rightarrow \psi(\sigma_2^m)$. Recuerde que un algoritmo basado en comparaciones produce las mismas salidas a entradas que guardan un orden equivalente. Por lo tanto, K^n tiene estructura binaria simétrica bajo mapeos con preservación de orden.

Por otra parte, no es difícil de verificar que la imagen dividida de las ejecuciones IS en las que los procesadores están activos una sola vez, no tiene aristas de cruce (ver Figura 3.11). En un algoritmo en el que los procesadores están activos más de una vez, simplemente la subdivisión será más fina. Por lo tanto, podemos decir que K^n no tiene aristas de cruce.

En la Figura 3.12 podemos sustituir el nombre decidido por su paridad para verificar que la imagen dividida de las ejecuciones del algoritmo *AlgRenombrado1* cumple todo lo anterior.

Finalmente, K^n es una imagen dividida cromática, orientable y conexa de σ^n bajo ψ , que además tiene estructura binaria simétrica y no tiene aristas de cruce. En el capítulo 4 estudiaremos este tipo de imágenes divididas.

Capítulo 4

Caracterización del Número de Simplejos Monocromáticos

En éste capítulo presentamos una caracterización combinatoria del número de simplejos monocromáticos en una imagen dividida cromática, orientable, conexa, con estructura binaria simétrica y sin aristas de cruce. Comenzamos dando un vistazo, sección 4.1, a la estrategia que seguiremos a lo largo del capítulo. En la sección 4.2 mostramos la forma en que contaremos los simplejos monocromáticos a través de la Generalización del lema del Índice. Las secciones 4.3 y 4.4 muestran un proceso inductivo sobre la imagen dividida, el cual a la postre nos dará la caracterización. Finalmente, la sección 4.5 presenta una imagen dividida de dimensión 5 que tiene una coloración binaria tal, que su índice es cero. Como veremos en el capítulo 5, todo esto tiene importantes implicaciones con respecto al problema del renombrado.

4.1. Estrategia de la Demostración

La estrategia que seguiremos a lo largo del capítulo es la siguiente. Primero agregaremos una tercera coloración c a la imagen dividida, de forma tal que un n -simplejo estará coloreado propiamente bajo c si y solo si es monocromático bajo la coloración binaria b . Esta propiedad nos permitirá contar los simplejos monocromáticos a través de la Generalización del Lema del Índice. Por lo tanto, en adelante lo que nos interesará saber, es que forma tiene el índice de la imagen dividida con respecto a la coloración c .

Ahora bien, para determinar la forma que tiene el índice, aplicaremos una suerte de proceso inductivo sobre la frontera de la imagen dividida, el cual funciona a grosso modo de la siguiente manera. Comenzaremos estableciendo la coloración binaria en la frontera igual a 0 (todo vértice en la frontera tiene color binario 0). Después, modificaremos grupos de vértices de la frontera (su color binario pasa de 0 a 1) con *carriers* de la misma dimensión, hasta obte-

ner la coloración original. A esta acción, modificar vértices con carriers de la misma dimensión, la conoceremos como *paso*. Una de las propiedades que presentarán estos pasos, es que después de aplicar cada uno de ellos, la simetría de la coloración binaria se preserva. Sin embargo, estas acciones no se realizarán de forma arbitraria, sino por dimensión. Esto es, se llevará a cabo un paso en el que se modifican vértices con carriers de dimensión $i + 1$ si y solo si ya no quedan por realizarse pasos que modifican vértices con carriers de dimensión i .

Existen dos aspectos importantes relacionados con los pasos en el proceso. El primero es la cantidad de vértices que se modifican en cada paso ellos. Este número está determinado directamente por la dimensión i de los carriers que se modifican en él, y tiene mucho que ver con el número de caras de dimensión i de un n -simplejo. El segundo aspecto es la forma en que afectan al índice cada una de las modificaciones que se realizan dentro del paso. La parte crucial de la demostración es que todas estas modificaciones afectan de la misma forma al índice, lo cual nos permitirá obtener la caracterización deseada.

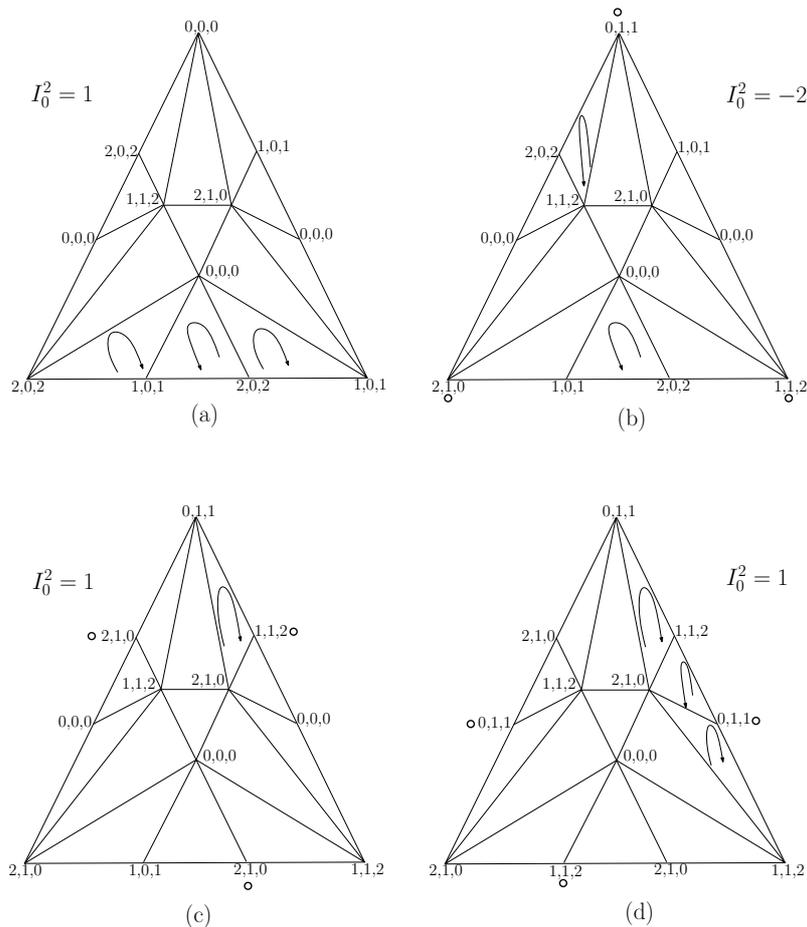


Figura 4.1: Un ejemplo del proceso inductivo.

La figura 4.1 muestra un ejemplo muy sencillo y descriptivo de nuestra estrategia. En esta se aprecia el proceso inductivo sobre una imagen dividida de un 2-simplejo. Los vértices de la imagen dividida tienen asociadas las coloraciones id , b y c , en ese orden. Esta última está definida como $c(v) = id(v) + b(v) \pmod 3$. Por simple inspección podemos verificar que un 2-simplejo es monocromático bajo b si y solo si está coloreado propiamente bajo c .

En cada uno de los pasos del proceso se muestra el índice I_0^2 con respecto a c , asumiendo que los 2-simplejos de la imagen dividida están orientados en el sentido inverso a las manecillas del reloj. Además, los vértices modificados en el paso se marcan con un pequeño círculo. El proceso comienza (a) con la coloración binaria en la frontera igual a 0 y con el índice igual a 1. Como veremos en las siguientes secciones, el índice al principio de proceso siempre es ± 1 , dependiendo de la orientación. En un primer paso (b), los vértices de las esquinas se modifican, es decir, su color binario pasa de 0 a 1. Este cambio resta 3 al índice que se tenía. Este número no es casualidad, pues está íntimamente ligado al número de vértices que se modificaron. En un segundo paso (c), tres vértices entre las esquinas se modifican. Estas modificaciones suman 3 al índice anterior. Finalmente, un tercer paso se lleva a cabo (d). Observe que en todo momento se mantiene la simetría de la coloración binaria y, además, el número de 2-simplejos monocromáticos contados por orientación es igual al índice.

4.2. Contando Simplejos Monocromáticos

Como se mencionó en la sección anterior, contaremos los simplejos monocromáticos de la imagen dividida a través de la Generalización del Lema del Índice. A continuación presentamos como logramos esto.

Considere una función f de $ID^n = \{0, 1 \dots n\}$ a ID^n .

Definición 4.2.1 *Decimos que la función f es una permutación mínima de ID^n , si f es una permutación de ID^n y la restricción $f|_I$ de cualquier subconjunto propio de ID^n no es una permutación de I .*

Observe que lo anterior implica que para todo $I \subset ID^n$, existe $x \in I$ tal que $f(x) \in ID^n - I$. En particular, para todo $x \in ID^n$, $f(x) \neq x$ cuando $n \geq 1$. Esta es una forma alternativa de definir un $(n + 1)$ -ciclo [2, capítulo 6].

Sea K^n un complejo con una coloración binaria b y otra propia id con ID^n . Añadiremos a K^n una tercera coloración c que nos ayudará a contar sus n -simplejos monocromáticos bajo b .

Definición 4.2.2 *Para una permutación mínima f de ID^n , sea*

$$c(v) = \begin{cases} id(v) & \text{si } b(v) = 0 \\ f(id(v)) & \text{si } b(v) = 1 \end{cases}$$

El siguiente lema muestra la relación entre las coloraciones b y c .

Lema 4.2.3 *Un n -simplejo de K^n es monocromático bajo b si y solo si está coloreado propiamente bajo c .*

Demostración: Advierta que el lema se cumple si $n = 0$, por tanto, asumiremos que $n \geq 1$. Primero observe que todo n -simplejo de K^n está coloreado propiamente bajo id . Por la definición de c , Definición 4.2.2, tenemos que si un n -simplejo de K^n es 0-monocromático bajo b , entonces está coloreado propiamente bajo c . Además, ya que f es una permutación mínima de ID^n , todo n -simplejo de K^n que es 1-monocromático bajo b , está coloreado propiamente bajo c .

Para demostrar la otra dirección, supongamos, por contradicción, que un n -simplejo τ coloreado propiamente bajo c , no es monocromático bajo b . Considere la restricción $id|_\tau$. Observe que $id|_\tau$ es una biyección. Sea id^{-1} la función inversa de $id|_\tau$, y C_i el subconjunto de ID^n tal que para todo $x \in C_i$, $b(id^{-1}(x)) = i$. Note que para todo $x \in C_0$, $c(id^{-1}(x)) = x$ y para todo $y \in C_1$, $c(id^{-1}(y)) = f(y)$. Dado que f es una permutación mínima de ID^n , existe un elemento w de C_1 tal que $f(w) \in ID - C_1 = C_0$. Por lo tanto, existe un elemento z de C_0 tal que $c(id^{-1}(w)) = c(id^{-1}(z))$. Una contradicción.

□

¿Que pasa con la coloración c de todos los otros simplejos de K^n ?

Lema 4.2.4 *Para $m < n$, un m -simplejo $\tau \in K^n$ está coloreado propiamente bajo c con $id(\tau)$ si y solo si es 0-monocromático bajo b .*

Demostración: El argumento es similar al usado en la demostración del Lema 4.2.3. Recuerde que τ está coloreado propiamente bajo id . Advierta que si τ es 0-monocromático, entonces para todo $v \in \tau$, $c(v) = id(v)$, y, por tanto, $c(\tau) = id(\tau)$. Ahora supongamos, por contradicción, que $c(\tau) = id(\tau)$ pero τ no es 0-monocromático. Considere la restricción $id|_\tau$. Sea id^{-1} la función inversa de $id|_\tau$, y C_i el subconjunto de ID^n tal que para todo $x \in C_i$, $b(id^{-1}(x)) = i$. Observe que $f|_{C_1}$ es una permutación de C_1 , ya que τ está coloreado propiamente bajo c con $id(\tau)$. Una contradicción.

□

De la definición de índice y contenido, Definición 3.1.3, y del Lema 4.2.3, obtenemos el siguiente corolario, en el cual I^n es el índice de K^n con respecto a c .

Corolario 4.2.5 *Si K^n es una pseudovariedad orientada coherentemente e $I^n \neq 0$, entonces K^n contiene por lo menos un n -simplejo monocromático bajo b .*

Demostración: Por la Generalización del Lema del Índice 3.1.5, si $I^n \neq 0$ entonces $C^n \neq 0$. De la definición de contenido, Definición 3.1.3 (1), podemos deducir que si $C^n \neq 0$, entonces K^n contiene por lo menos un n -simplejo coloreado propiamente bajo c . Además, por el Lema 4.2.3, un n -simplejo de K^n está coloreado propiamente bajo c si y solo si es monocromático bajo b . Por lo tanto, K^n contiene por lo menos un n -simplejo monocromático bajo b .

□

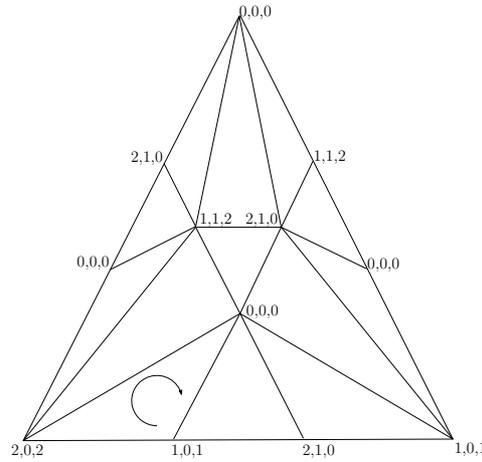


Figura 4.2: Una pseudovariación con las tres coloraciones.

La Figura 4.2 muestra una 2-pseudovariación cromática y orientable con las tres coloraciones id , b y c , en ese orden, asociadas a cada uno de sus vértices. La coloración c ocupa la permutación mínima $f(x) = (x+1) \bmod 3$. El simplejo con la flecha circular es el único simplejo monocromático y, por tanto, está coloreado propiamente bajo c . El índice I_2^2 con respecto a c es igual a -1 , asumiendo que cada 2-simplejo está orientado en el sentido inverso a las manecillas del reloj. Observe que el 2-simplejo que está coloreado propiamente bajo c es contado como -1 por C^2 .

El Lema 4.2.3 implica que el contenido de K^n con respecto a c cuenta los n -simplejos monocromáticos de K^n . Ahora presentamos un lema acerca de cómo los n -simplejos b -monocromáticos de K^n son contados por el contenido. Básicamente, lo que el lema dice es que un n -simplejo 0 -monocromático con orientación d es contado como d , y un n -simplejo 1 -monocromático con orientación d es contado como d si n es par y como $-d$ si es impar. Recuerde que $d \in \{+1, -1\}$ y $b \in \{0, 1\}$.

Lema 4.2.6 *Sea K^n una pseudovariación cromática y orientada coherentemente, que además tiene una coloración binaria b . Considere un n -simplejo b -monocromático τ de K^n con orientación d . El simplejo τ es contado como $(-1)^{b*n}d$ por el contenido de K^n con respecto a c .*

Demostración: Considere la secuencia $S = \langle 0, 1 \dots n \rangle$ de ID^n . Lo que buscamos saber es si S induce ordenes equivalentes a los vértices de τ con respecto a id y c , respectivamente. Recuerde que S pertenece a la orientación de un n -simplejo de K^n con respecto a id , si y solo si este tiene orientación $+1$.

Primero supongamos que τ es 0-monocromático. Por la definición de c , Definición 4.2.2, para todo vértice $v \in \tau$, $c(v) = id(v)$. Es decir, S induce el mismo orden a los vértices de τ con respecto de id y c , respectivamente. Por la definición de contenido, Definición 3.1.3 (1), τ es contado como $(-1)^{0*n}d = d$.

Ahora supongamos que τ es 1-monocromático. Por la definición de c , Definición 4.2.2, para todo vértice $v \in \tau$, $c(v) = f(id(v))$. Sea S' la secuencia de ID^n tal que $S'_i = f(S_i)$. Observe que S representa la coloración id y S' representa la coloración c . Sean S_{id} y S_c las secuencias de τ inducidas por S y S' con respecto a id y c , respectivamente. Ahora aplicamos una sucesión de n transposiciones sobre S' , empezando con $j = 0$ hasta $j = n - 1$, donde la j -ésima transposición consiste en intercambiar S'_j y $S'_k = j$, $j \leq k \leq n$. Es decir, después de la l -ésima transposición, los primeros $l + 1$ elementos en la secuencia están ordenados ascendentemente del 0 a l . Por lo tanto, $S = S'$ después de aplicar las n transposiciones. Observe que si n es par, entonces S_c es una transposición par de S_{id} y, por ende, S_c pertenece a la orientación de σ^n si y solo si S_{id} pertenece a la orientación de σ^n (recuerde que la orientación de un simplejo es una secuencia de sus vértices junto con todas sus transposiciones pares). Por la definición de contenido, Definición 3.1.3 (1), τ es contado como $(-1)^{1*n}d = d$. Ahora bien, si n es impar, entonces S_c es una transposición impar de S_{id} y, por tanto, S_c pertenece a la orientación de σ^n si y solo si S_{id} no pertenece a la orientación de σ^n . Es decir, τ es contado como $(-1)^{1*n}d = -d$.

Advierta que en el caso anterior asumimos que todas las n transposiciones aplicadas sobre S' fueron distintas a la transposición identidad. A continuación demostramos esto. Primero recuerde que f es una transposición mínima de ID^n . Supongamos, pro contradicción, que $S'_j = j$ antes de la j -ésima transposición. Es decir, el color j está en su lugar antes de lo esperado. Considere la primera transposición i , $i < j$, después de la cual $S'_j = j$. Observe que antes de aplicar la i -ésima transposición tenemos que $S'_i = j$ y $S'_j = i$. Por lo tanto, la restricción $f|_{\{i,j\}}$ es una transposición de $\{i, j\}$. Una contradicción. Esto completa la demostración.

□

4.3. Caracterización Combinatoria

En esta sección presentaremos el proceso inductivo descrito informalmente en la sección 4.1. Con este fin, fijamos una imagen dividida K^n cromática, conexa y orientada coherentemente de σ^n bajo ψ , con estructura binaria simétrica y sin aristas de cruce. También asumimos que K^n tiene una tercera coloración c como la que presenta en la Definición 4.2.2. En los subsecuente, cuando mencionemos que un simplejo de K^n tiene una coloración propia, nos referimos a la coloración c , a menos que se especifique otra. Si decimos que es monocromático,

entonces nos referimos a la coloración b . Además que siempre contaremos los simplejos coloreados propiamente de K^n como en la definición de contenido, Definición 3.1.3 (1). Y el índice y contenido de K^n siempre serán calculados con respecto a c . El siguiente lema es la base de nuestro proceso inductivo.

Lema 4.3.1 *Si para todo vértice v de $bd(K^n)$, $b(v) = 0$, entonces $I_i^n = \pm 1$, dependiendo de la orientación.*

Demostración: Por la definición de c , Definición 4.2.2, para todo vértice v de $bd(K^n)$, $c(v) = id(v)$. Considere las caras $\sigma_1^0, \sigma_2^1 \dots \sigma_n^{n-1}$ de σ^n tales que $\sigma_1^0 \subset \sigma_2^1 \dots \subset \sigma_n^{n-1}$. Recuerde que σ_i^{i-1} es la cara de σ_{i+1}^i sin el vértice coloreado i . Por lo tanto, $ID(\sigma_1^0) = \{0\}$, $ID(\sigma_2^1) = \{0, 1\}$, y así sucesivamente. Puesto que K^n es una imagen dividida cromática, tenemos que $\psi(\sigma_n^{n-1})$ contiene todos los $(n-1)$ -simplejos de $bd(K^n)$ coloreados propiamente con ID_n^n . De hecho, todo $(n-1)$ -simplejo de $\psi(\sigma_n^{n-1})$ está coloreado propiamente con ID_n^n . Considere el complejo $K^{n-1} = \psi(\sigma_n^{n-1})$. Por el Lema 3.1.8 (3), K^{n-1} es una imagen dividida de σ_n^{n-1} bajo $\psi|_{M(\sigma_n^{n-1})}$. Por definición, K^n es cromática, orientable y conexa. Aún más, por el Lema 3.1.17, K^n induce una orientación coherente a K^{n-1} . Por el Lema 3.1.6, tenemos que $I_n(K^n) = (-1)^{n-1}I_{n-1}(K^{n-1})$. Observe que podemos construir el subcomplejo $K^{n-2} = \psi(\sigma_{n-1}^{n-2})$ tal que $I_{n-1}(K^{n-1}) = (-1)^{n-2}I_{n-2}(K^{n-2})$, y así sucesivamente. Es decir, para $0 < m \leq n-1$, $I_m(K^m) = (-1)^{m-1}I_{m-1}(K^{m-1})$, donde $K^m = \psi(\sigma_{m+1}^m)$ y $K^{m-1} = \psi(\sigma_m^{m-1})$. De esta forma, $I_n(K^n) = (-1)^{n-1}(-1)^{n-2} \dots (-1)^1 I_1(K^1)$. Además, por el Lema 3.1.6, sabemos que $I_1(K^1) = C(K^0)$, donde $K^0 = \psi(\sigma_1^0)$. Advierta que K^0 contiene únicamente un 0-simplejo coloreado 0. Es decir, $C(K^0) = \pm 1$, dependiendo de su orientación, y, por tanto, $I_n(K^n) = \pm 1$. Finalmente, por la Generalización del Lema del Índice 3.1.5, podemos concluir que $(-1)^i I_i(K^n) = (-1)^n I_n(K^n)$.

□

A continuación introducimos una notación más. Sea L un complejo coloreado e I un conjunto con m colores. Denotamos como $k^m(L, I)$ al número de los m -simplejos de L que tienen una coloración propia con los colores en I . Algunas veces nos referiremos a este valor como el valor k de L con I o simplemente el valor k de L .

En lo que resta del capítulo, frecuentemente calcularemos el valor k de $st(v, L)$ y $st(v_1, L) \cup st(v_2, L) \cup \dots \cup st(v_q, L)$, respectivamente. Con el fin de hacer nuestra notación menos complicada, simplemente escribiremos $k^m(v, L, I)$ y $k^m(v_1, v_2, \dots, v_n, L, I)$, respectivamente. En la Figura 4.3 se puede ver una imagen dividida cromática K^2 de un 2-simplejo σ^2 bajo ψ . Observe que $st(v, bd(K^2)) = st(v, \psi(\sigma_2^1)) \cup st(v, \psi(\sigma_1^1))$ y $k^1(st(v, bd(K^2)), \{0, 1\}) = -1$, asumiendo que todos los 2-simplejos de K^2 están orientados en el sentido inverso a las manecillas del reloj.

Como se mencionó en la sección 4.1, durante el proceso inductivo modificaremos la coloración binaria de los vértices en la frontera. Considere un vértice

$v \in bd(K^n)$ tal que $b(v) = 0$. Decimos que el vértice v ha sido *procesado* si cambiamos su coloración binaria de 0 a 1. Advierta que esto provoca un cambio en la coloración c . Denotamos como $b(v)$ y $c(v)$ a los colores de v antes del cambio, y como $\dot{b}(v)$ y $\dot{c}(v)$ a los colores después del mismo. De forma similar, I_i^n es el índice de K^n antes de procesar v e \dot{I}_i^n el índice después de procesarlo. En general, marcamos con un punto en la parte superior a los valores, coloraciones y simplejos después de procesar el vértice v .

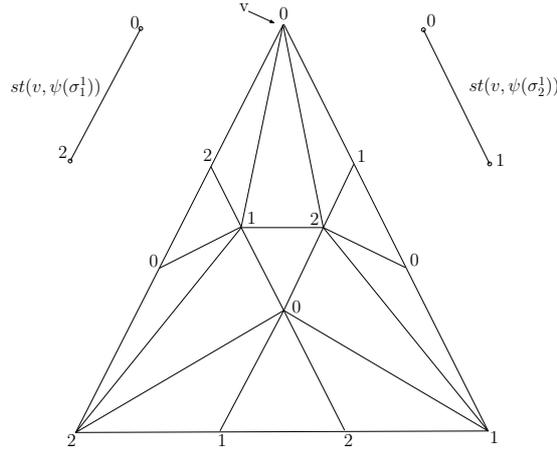


Figura 4.3: Complejo estrella de un vértice en la frontera.

El siguiente lema muestra la forma en que el índice cambia cuando procesamos un vértice en la frontera. Recuerde que cuando decimos que un simplejo de K^n está coloreado propiamente, nos referimos a la coloración c . Por lo tanto, si L es un subcomplejo de K^n , $I \subseteq ID^n$ e $|I| = m$, entonces $k^m(L, I)$ son los m -simplejos de L que están coloreados propiamente bajo c con I .

Lema 4.3.2 *Considere un vértice v de $bd(K^n)$ tal que $b(v) = 0$. Si v es procesado, entonces*

$$\dot{I}_i^n = I_i^n + \dot{k}^{n-1}(v, bd(K^n), ID_i^n) - k^{n-1}(v, bd(K^n), ID_i^n)$$

Demostración: Considere un $(n-1)$ -simplejo τ^{n-1} de $st(v, bd(K^n))$. Tenemos dos casos. Si $c(\tau^{n-1}) = ID_i^n$, entonces $c(\dot{\tau}^{n-1}) \neq ID_i^n$. Recuerde que f es una permutación mínima y, por tanto, $c(v) \neq \dot{c}(v)$. Ahora, si $c(\tau^{n-1}) \neq ID_i^n$, entonces podría ser el caso en que $c(\dot{\tau}^{n-1}) = ID_i^n$. En palabras, el cambio en v provoca que algunos $(n-1)$ -simplejos que no estaban coloreados propiamente con ID_i^n , después del cambio lo estén, y otros que si lo estaban, lo dejen de estar. Además, advierta que $st(v, bd(K^n))$ contiene todos los $(n-1)$ -simplejos de $bd(K^n)$ que cambian su coloración c cuando v es procesado. De la definición de índice, Definición 3.1.3 (2), sabemos que I_i^n cuenta todos los $(n-1)$ -simplejos de $bd(K^n)$ que están coloreados propiamente con ID_i^n . Por lo tanto, podemos afirmar que \dot{I}_i^n es I_i^n más los $(n-1)$ -simplejos en $bd(K^n)$ que adquieren una coloración propia con ID_i^n después de procesar v , menos los $(n-1)$ -simplejos en

$bd(K^n)$ que dejan de tener una coloración propia con ID_i^n después del cambio. Es decir

$$\dot{I}_i^n = I_i^n + k^{n-1}(v, bd(K^n), ID_i^n) - k^{n-1}(v, bd(K^n), ID_i^n)$$

□

Considere las caras $\sigma_1^0, \sigma_2^1 \dots \sigma_n^{n-1}$ de σ^n tales que $\sigma_1^0 \subset \sigma_2^1 \subset \dots \subset \sigma_n^{n-1}$. Recuerde que σ_i^{i-1} es la cara de σ_{i+1}^i sin el vértice coloreado i . Por lo tanto, $ID(\sigma_1^0) = \{0\}$, $ID(\sigma_2^1) = \{0, 1\}$, y así sucesivamente. Fijaremos estas caras por el resto de la sección. Además, asumiremos que $\psi(\sigma_i^{i-1})$ tiene la orientación inducida por $\psi(\sigma_{i+1}^i)$, $0 \leq i \leq n-1$, y $\psi(\sigma_n^{n-1})$ tiene la orientación inducida por K^n . Por el Lema 3.1.17, todas estas orientaciones son coherentes. El siguiente lema muestra una forma alterna de calcular $k^{n-1}(v, bd(K^n), ID_n^n)$ para un caso particular.

Lema 4.3.3 *Considere la cara σ_{l+1}^l . Sea v un vértice de $\psi(\sigma_{l+1}^l)$ tal que $b(v) = 0$. Si para $l < p < n$, para todo vértice interno $u \in \psi(\sigma_{p+1}^p)$, $b(u) = 0$, entonces*

$$k^{n-1}(v, bd(K^n), ID_n^n) = (-1)^e k^l(v, \psi(\sigma_{l+1}^l), ID_I^n)$$

donde $e = ((l+1) + (l+2) + \dots + (n-1))$ e $I = \{l+1, l+2 \dots n\}$.

Consideramos que la idea del proceso inductivo es más clara si presentamos la demostración del lema anterior en la siguiente sección, ya que esta es un poco larga y necesita algunos lemas y definiciones adicionales. A continuación presentamos un corolario que se obtiene directamente del Lema 4.3.3.

Corolario 4.3.4 *Considere la cara σ_{l+1}^l . Sean v_1, v_2, \dots, v_q vértices de $\psi(\sigma_{l+1}^l)$ tales que $b(v_i) = 0$, $1 \leq i \leq q$. Si para $l < p < n$, para todo vértice interno $u \in \psi(\sigma_{p+1}^p)$, $b(u) = 0$, entonces*

$$k^{n-1}(v_1, v_2, \dots, v_q, bd(K^n), ID_n^n) = (-1)^e k^l(v_1, v_2, \dots, v_q, \psi(\sigma_{l+1}^l), ID_I^n)$$

donde $e = ((l+1) + (l+2) + \dots + (n-1))$ e $I = \{l+1, l+2 \dots n\}$.

Ahora presentamos un lema acerca del número de vértices en la frontera que deben ser procesados para preservar la simetría de la coloración binaria. Para esto, tenga en mente que un n -simplejo tiene $\binom{n+1}{l+1}$ caras de dimensión l .

Lema 4.3.5 *Si procesamos un vértice con carrier de dimensión l , $0 \leq l \leq n$, entonces tenemos que procesar otros $\binom{n+1}{l+1} - 1$ vértices con carriers de dimensión l para preservar la simetría de la coloración binaria.*

Demostración: Sean $\sigma_1^l, \sigma_2^l \dots \sigma_m^l$ las l -caras de σ^n , $m = \binom{n+1}{l+1}$. Para $1 \leq k \leq m$, sea f_k la biyección simplicial con preservación de orden de $\psi(\sigma_1^l)$ a $\psi(\sigma_k^l)$.

Considere las caras σ_i^l y σ_j^l . Observe que $f_j \circ f_i^{-1}$ es la biyección con preservación de orden de $\psi(\sigma_i^l)$ a $\psi(\sigma_j^l)$. Considere dos p -caras σ_i^p y σ_j^p de σ^n tales que $\sigma_i^l \subset \sigma_i^p$, $\sigma_j^l \subset \sigma_j^p$ y $\psi(\sigma_j^l)$ es el rango de $\dot{f}|_{\psi(\sigma_i^l)}$, donde \dot{f} es la biyección con preservación de orden de $\psi(\sigma_i^p)$ a $\psi(\sigma_j^p)$. Por la Afirmación 3.1.19, tenemos que $\dot{f}|_{\psi(\sigma_i^l)} = f_j \circ f_i^{-1}$ y $\dot{f}|_{\psi(\sigma_i^l)}^{-1} = f_i \circ f_j^{-1}$. Por lo tanto, si procesamos un vértice $v \in \psi(\sigma_1^l)$, entonces tenemos que procesar sus vértices isomorfos $f_k(v) \in \psi(\sigma_k^l)$, para preservar la simetría de la coloración binaria.

□

Para $0 \leq l \leq n-1$, sean $\sigma_{l+1}^l = \tau_1^l, \tau_2^l \dots \tau_m^l$ todas las l -caras de σ^n . Sea f_k , $1 \leq k \leq m$, la biyección con preservación de orden de $\psi(\tau_1^l)$ a $\psi(\tau_k^l)$. Recuerde que la biyección f_k es única (ver sección 3.1.4). Considere un vértice interno v de $\psi(\tau_1^l)$ tal que $b(v) = 0$. Sean $v_1, v_2 \dots v_m$ los vértices de K^n tales que $v_k = f_k(v)$, $1 \leq k \leq m$. Primero observe que $v = v_1$. Además, v_k es un vértice interno de $\psi(\tau_k^l)$ con $b(v_k) = 0$. Aún más, por la Afirmación 3.1.10, $v_1, v_2 \dots v_m$ tienen carriers de dimensión l .

Dentro del proceso inductivo, un l -paso consiste en procesar uno por uno, en cualquier orden, los vértices $v_1, v_2 \dots v_m$. De esta forma, el proceso comienza con una coloración binaria en la frontera igual a 0, es decir, para todo vértice v en $bd(K^n)$, $b(v) = 0$, y termina cuando todos los vértices en $bd(K^n)$ tienen los colores originales. La forma en que se modifican los vértices es a través de l -pasos. Sin embargo, estos no se realizan de forma arbitraria, sino por dimensiones. A saber, un $(l+1)$ -paso es llevado a cabo si y solo si todos los l -pasos necesarios han sido realizados. En otras palabras, un vértice interno de $\psi(\sigma_{l+1}^l)$ es procesado si y solo si todos los vértices en $\psi(\sigma_{l+1}^{l-1})$ tienen su configuración original. Observe que lo anterior implica cuando un vértice interno de $\psi(\sigma_{l+1}^l)$ es procesado, es por que $bd(\psi(\sigma_{l+1}^l))$ ya tiene su configuración original. Por el resto de la sección fijaremos los vértices $V^l = \{v_1 \dots v_m\}$ y el l -paso asociado a ellos. Advierta que dado a que en el proceso inductivo los pasos se realizan por dimensiones, tenemos que para todo vértice interno v de $\psi(\sigma_{p+1}^p)$, $m < p < n$, $b(v) = 0$.

De forma similar que cuando procesamos un vértice, marcamos con el símbolo $\hat{\cdot}$ a los valores, colores y simplejos después de un l -paso. El siguiente lema enseña la relación que guardan los valores k de los vértices en V^l .

Lema 4.3.6 *Considere los vértices $v_f, v_g, v_h \in V^l$ tales que $v_f, v_g \in \psi(\sigma_n^{n-1})$ y $v_h \in \psi(\sigma_i^{n-1})$, donde $\sigma_i^{n-1} \subset \sigma^n$. Entonces*

$$\begin{aligned} k^{n-1}(v_f, bd(K^n), ID_n^n) &= k^{n-1}(v_g, bd(K^n), ID_n^n) \\ (-1)^n k^{n-1}(v_f, bd(K^n), ID_n^n) &= (-1)^i k^{n-1}(v_h, bd(K^n), ID_i^n) \end{aligned}$$

Demostración: Procederemos por inducción en n . La base de la inducción es para la cara σ_{l+2}^{l+1} . Note que $ID(\sigma_{l+2}^{l+1}) = ID_I^n$, donde $I = \{l+2, l+3 \dots n\}$.

Denotaremos a ID_I^n como ID' y a $\psi(\sigma_{l+2}^{l+1})$ como K^{l+1} . De acuerdo al lema, $v_f \in \psi(\sigma_{l+1}^l)$ y $v_h \in \psi(\sigma_i^l)$, donde σ_i^l es una cara de σ_{l+2}^{l+1} . Por la Afirmación 3.1.10, v_f y v_h son vértices internos de $\psi(\sigma_{l+1}^l)$ y $\psi(\sigma_i^l)$, respectivamente. Por lo tanto, $st(v_f, bd(K^{l+1})) = st(v_f, \psi(\sigma_{l+1}^l))$ y $st(v_h, bd(K^{l+1})) = st(v_h, \psi(\sigma_i^l))$.

Sean τ_1^l y τ_2^l simplejos de $st(v_f, \psi(\sigma_{l+1}^l))$ y $st(v_h, \psi(\sigma_i^l))$, respectivamente, tales que son isomorfos (recuerde que nos referimos a que lo son bajo la biyección simplicial con preservación de orden). Advierta que $id(\tau_1^l) = ID'_{l+1}$ e $id(\tau_2^l) = ID'_i$. Aún más, para todo par de vértices isomorfos $u \in \tau_1^l$ y $v \in \tau_2^l$, tenemos que $b(u) = b(v)$. Por el Lema 4.2.4, $c(\tau_1^l) = ID'_{l+1}$ si y solo si τ_1^l es 0-monocromático. De forma similar ocurre con τ_2^l e ID'_i . Por lo tanto, $c(\tau_1^l) = ID'_{l+1}$ si y solo si $id(\tau_2^l) = ID'_i$. Por el Lema 3.1.22, τ_1^l y τ_2^l tienen las mismas orientaciones, multiplicadas por $(-1)^{l+1}$ y $(-1)^i$, respectivamente. Esto es

$$(-1)^{l+1}k^l(v_f, bd(K^{l+1}), ID'_{l+1}) = (-1)^i k^l(v_h, bd(K^{l+1}), ID'_i)$$

Observe que para éste caso, $\psi(\sigma_{l+1}^l)$ y $\psi(\sigma_i^l)$ no contienen mas vértices que vayan a ser procesados en el l -paso.

Supongamos que el lema se cumple para l' . Demostraremos que se sigue cumpliendo para $l' + 1$. De nuevo, de acuerdo al lema, $v_f \in \psi(\sigma_{l'+1}^{l'})$ y $v_h \in \psi(\sigma_i^{l'})$, donde $\sigma_i^{l'}$ es una cara de $\sigma_{l'+2}^{l'+1}$. Note que $ID(\sigma_{l'+2}^{l'+1}) = ID_I^n$, donde $I = \{l'+2, l'+3 \dots n\}$. Denotaremos a ID_I^n como ID' y a $\psi(\sigma_{l'+2}^{l'+1})$ como $K^{l'+1}$. Por el Lema 4.3.3

$$k^{l'}(v_f, bd(K^{l'+1}), ID'_{l'+1}) = k^{l'}(v_f, \psi(\sigma_{l'+1}^{l'}), ID'_{l'+1})$$

y

$$k^{l'}(v_h, bd(K^{l'+1}), ID'_i) = k^{l'}(v_h, \psi(\sigma_i^{l'}), ID'_i)$$

Utilizando el mismo argumento que en el caso base, podemos probar lo siguiente

$$(-1)^{l'+1}k^{l'}(v_f, bd(K^{l'+1}), ID'_{l'+1}) = (-1)^i k^{l'}(v_h, bd(K^{l'+1}), ID'_i)$$

Ahora, recuerde que v_f y v_h son vértices con carriers de dimensión l' . Por lo tanto, $\psi(\sigma_{l'+1}^{l'})$ contiene $\binom{l'+1}{l'}$ vértices de $V^{l'}$. Sea v_g el vértice de $V^{l'}$ que necesita el lema. Es decir, $v_g \in \psi(\sigma_{l'+1}^{l'})$. Considere una cara $\sigma_j^{l'-1}$ de $\sigma_{l'+1}^{l'}$ tal que $v_g \in \psi(\sigma_j^{l'-1})$. Supongamos, s.p.g., que $v_f \in \sigma_{l'}^{l'-1}$. Por el Lema 4.3.3

$$k^{l'}(v_f, bd(K^{l'+1}), ID'_{l'+1}) = (-1)^{l'}k^{l'-1}(v_f, \psi(\sigma_{l'}^{l'-1}), ID'_{\{l', l'+1\}})$$

y

$$k^{l'}(v_g, bd(K^{l'+1}), ID'_{l'+1}) = (-1)^j k^{l'-1}(v_g, \psi(\sigma_j^{l'-1}), ID'_{\{j, l'+1\}})$$

Todavía más, por hipótesis de inducción

$$(-1)^{l'}k^{l'-1}(v_f, bd(\psi(\sigma_{l'}^{l'-1})), ID'_{\{l', l'+1\}}) = (-1)^j k^{l'-1}(v_g, bd(\psi(\sigma_j^{l'-1})), ID'_{\{j, l'+1\}})$$

Además, por el Lema 4.3.3

$$k^{l'-1}(v_f, bd(\psi(\sigma'_{l'+1})), ID'_{\{l', l'+1\}}) = k^{l'-1}(v_f, \psi(\sigma'^{l'-1}), ID'_{\{l', l'+1\}})$$

y

$$k^{l'-1}(v_g, bd(\psi(\sigma'_{l'+1})), ID'_{\{j, l'+1\}}) = k^{l'-1}(v_g, \psi(\sigma'_j{}^{l'-1}), ID'_{\{j, l'+1\}})$$

Por lo tanto

$$k^{l'}(v_f, bd(K^{l'+1}), ID'_{l'+1}) = k^{l'}(v_g, bd(K^{l'+1}), ID'_{l'+1})$$

Esto completa la demostración. □

A continuación presentamos un par de lemas que muestran como los l -pasos del proceso afectan el índice de K^n .

Lema 4.3.7 *El índice de K^n después del l -paso asociado a V^l , es $\hat{I}_i^n = I_i^n - \binom{n+1}{l+1}k$, donde k es el valor k de algún vértice de V^l que pertenezca a $\psi(\sigma_i^{n-1})$.*

Demostración: Recuerde que I_i^n es el índice antes del l -paso. Sea $v \in V^l$ tal que $v \in \psi(\sigma_i^{n-1})$, donde $\sigma_i^{n-1} \subset \sigma^n$. Por el Lema 4.3.2, cuando v es procesado

$$\hat{I}_i^n = I_i^n + \dot{k}^{n-1}(v, bd(K^n), ID_i^n) - k^{n-1}(v, bd(K^n), ID_i^n)$$

Primero demostraremos que $\dot{k}^{n-1}(v, bd(K^n), ID_i^n) = 0$. Tenemos los siguientes dos casos.

Si $l = n - 1$, entonces, por la Afirmación 3.1.10, v es interno. Por ende, $st(v, bd(K^n)) = st(v, \psi(\sigma_i^{n-1}))$. Es decir, todos los simplejos de $st(v, bd(K^n))$ son simplejos de $\psi(\sigma_i^{n-1})$. Considere un simplejo $\tau^{n-1} \in st(v, \psi(\sigma_i^{n-1}))$. Recuerde que $id(\tau^{n-1}) = ID_i^n$. Por el Lema 4.2.4, $c(\tau^{n-1}) = ID_i^n$ si y solo si τ^{n-1} es 0-monocromático. Advierta que $\dot{\tau}^{n-1}$ no es 0-monocromático. Por lo tanto, $c(\dot{\tau}^{n-1}) \neq ID_i^n$. Esto implica que $\dot{k}^{n-1}(v, bd(K^n), ID_i^n) = 0$.

Si $l < n - 1$, entonces, por la Afirmación 3.1.10, v es externo. Recuerde que para $l < q < n$, para todo vértice interno $v \in \psi(\sigma^q)$, $b(v) = 0$, donde $\sigma^q \subset \sigma^n$. Sea σ^l la cara de σ^n tal que v es vértice interno de $\psi(\sigma^l)$. Observe que $\sigma^l \subset \sigma_i^{n-1}$. Considere una cara σ_j^{n-1} de σ^n tal que $\sigma^l \subset \sigma_j^{n-1}$ e $i \neq j$. Tenemos que $v \in \psi(\sigma_j^{n-1})$ e $i \notin ID(\sigma^l)$. Considere un simplejo $\tau^l \in st(v, \psi(\sigma^l))$. Sea τ^{n-1} un simplejo de $st(v, \psi(\sigma_j^{n-1}))$ tal que $\tau^l \subset \tau^{n-1}$. Sea w el vértice de τ^{n-1} tal que $id(w) = i$ (recuerde que K^n es una imagen dividida cromática). Note que $w \notin \tau^l$. Por el Lema 3.1.13, existe una cara propia de σ^p de σ^n tal que w es un vértice interno de $\psi(\sigma^p)$ y, además, $\sigma^l \subset \sigma^p$. Observe que $c(w) = i$, ya que $b(w) = 0$. Por lo tanto, $c(\tau^{n-1}) \neq ID_i^n$ y $\dot{c}(\tau^{n-1}) \neq ID_i^n$. Es decir, $st(v, \psi(\sigma_i^{n-1}))$ contiene todos los $(n-1)$ -simplejos de $st(v, bd(K^n))$ que están coloreados propiamente con ID_i^n . Igual que en el caso anterior, por el Lema

4.2.4, un $(n-1)$ -simplejo de $\psi(\sigma_i^{n-1})$ está coloreado propiamente con ID_i^n si y solo si es 0-monocromático. Puesto que todo $(n-1)$ -simplejo de $st(v, \psi(\sigma_i^{n-1}))$ no es 0-monocromático, entonces $\hat{k}^{n-1}(v, bd(K^n), ID_i^n) = 0$.

Ahora probaremos el lema. Considere las caras σ_n^{n-1} y σ_j^{n-1} de σ^n . Sean v_g y v_h vértices de V^l tales que $v_g \in \psi(\sigma_n^{n-1})$ y $v_h \in \psi(\sigma_j^{n-1})$. Cuando v_h es procesado, el índice es

$$\hat{I}_j^n = I_j^n - k^{n-1}(v_h, bd(K^n), ID_j^n) \quad (4.1)$$

Todavía más, por la Generalización del Lema del Índice 3.1.5

$$(-1)^n I_n^n = (-1)^j I_j^n \quad (4.2)$$

y

$$(-1)^n \hat{I}_n^n = (-1)^j \hat{I}_j^n \quad (4.3)$$

Por (4.2) y (4.3),

$$\hat{I}_n^n = (-1)^p \hat{I}_j^n \quad (4.4)$$

y

$$I_n^n = (-1)^p I_j^n \quad (4.5)$$

donde $p = j - n$. Usando (4.1), (4.4) y (4.5)

$$\hat{I}_n^n = I_n^n - (-1)^p k^{n-1}(v_h, bd(K^n), ID_j^n) \quad (4.6)$$

Ahora, por el Lema 4.3.6

$$k^{n-1}(v_g, bd(K^n), ID_n^n) = (-1)^p k^{n-1}(v_h, bd(K^n), ID_j^n) \quad (4.7)$$

Por lo tanto, por (4.6) y (4.7)

$$\hat{I}_n^n = I_n^n - k^{n-1}(v_g, bd(K^n), ID_n^n)$$

Es decir, cuando v_h es procesado, el índice I_n^n cambia como si v_g hubiera sido procesado. De hecho, I_n^n cambia de la misma forma cuando es procesado cualquier vértice de V^l . Por otro lado, por el Lema 3.1.11, $v_h \notin st(v_g, bd(K^n))$. Por lo tanto, los $(n-1)$ -simplejos de $st(v_g, bd(K^n))$ no cambian su coloración. Esto es

$$k^{n-1}(v_g, bd(K^n), ID_n^n) = \hat{k}^{n-1}(v_g, bd(K^n), ID_n^n)$$

Por lo tanto, el índice después del l -paso es

$$\hat{I}_n^n = I_n^n - k \binom{n+1}{l+1}$$

donde $k = k^{n-1}(v_g, bd(K^n), ID_n^n)$. Por la Generalización del Lema del Índice 3.1.5, tenemos que $(-1)^n I_n^n = (-1)^i I_i^n$ y $(-1)^n \hat{I}_n^n = (-1)^i \hat{I}_i^n$. Además, por el Lema 4.3.6, $(-1)^n k = (-1)^i k'$, donde k' es el valor k de algún vértice de V^l que pertenezca a $\psi(\sigma_i^{n-1})$.

□

Observe que en el lema anterior, k representa el número de $(n-1)$ -simplejos de $\psi(\sigma_i^{n-1})$ que están coloreados propiamente con ID_i^n y que desaparecen cuando el l -paso es realizado. Por ejemplo, en el proceso que se muestra en la Figura 4.1, el valor k del vértice de $\psi(\sigma_0^1)$ que es procesado en el primer 1-paso, (c), es $+1$ (este valor se debe calcular en (b)). Al final del paso, el índice es $\hat{I}_0^2 = I_0^2 - 3 * 1$. El siguiente lema muestra como todos los l -pasos en el proceso afectan el índice.

Lema 4.3.8 *Supongamos que el proceso tiene un total de q l -pasos. Sea I_i^n el índice justo antes de realizar el primero de ellos. El índice después de todos los l -pasos es*

$$\hat{I}_i^n = I_i^n - \binom{n+1}{l+1} k_l$$

donde k_l es el valor k de los vértices v_1, v_2, \dots, v_q de $\psi(\sigma_i^{n-1})$ que fueron procesados en los l -pasos.

Demostración: Supongamos que v_i , $1 \leq i \leq q$, es procesado en el i -ésimo l -paso. Para $1 \leq p \leq q$, sea $I_{i,p}^n$ el índice después del p -ésimo l -paso. Por el Lema 4.3.7, los índices de los q l -pasos son los siguientes

$$\begin{aligned} I_{i,1}^n &= I_i^n - \binom{n+1}{l+1} k_1 \\ I_{i,2}^n &= I_{i,1}^n - \binom{n+1}{l+1} k_2 = I_i^n - \binom{n+1}{l+1} k_1 - \binom{n+1}{l+1} k_2 \\ &\vdots \\ I_{i,q}^n &= I_{i,q-1}^n - \binom{n+1}{l+1} k_q = I_i^n - \binom{n+1}{l+1} k_1 - \dots - \binom{n+1}{l+1} k_q \end{aligned}$$

donde $k_i = k^{n-1}(v_i, bd(K^n), ID_i^n)$, $1 \leq i \leq q$. Por lo tanto, $\hat{I}_i^n = I_i^n - \binom{n+1}{l+1} k_l$, donde $k_l = k_1 + k_2 + \dots + k_q$. Ahora, observe que k_l representa el número de $(n-1)$ -simplejos de $\psi(\sigma_i^{n-1})$ que están coloreados propiamente con ID_i^n y que desaparecerán cuando los l -pasos se hayan llevado a cabo. Todos estos simplejos están en $st(v_1, \psi(\sigma_i^{n-1})) \cup st(v_2, \psi(\sigma_i^{n-1})) \cup \dots \cup st(v_q, \psi(\sigma_i^{n-1}))$. Es decir, $k_l = k^{n-1}(v_1, v_2, \dots, v_q, bd(K^n), ID_i^n)$.

□

En el lema anterior, advierta que k_l debe calcularse antes de llevar a cabo alguno de los q l -pasos. Finalmente tenemos lo siguiente.

Lema 4.3.9 *Al final del proceso, el índice de K^n es*

$$I_i^n = \pm 1 - \sum_{j=0}^{n-1} \binom{n+1}{j+1} k_l$$

donde k_l es el valor k de los vértices internos de $\psi(\sigma_i^{n-1})$ que fueron procesados en los l -pasos del proceso.

Demostración: Por el Lema 4.3.1, el índice de K^n al comienzo del proceso es $I_{i,-1}^n = \pm 1$. Sea $I_{i,l}^n$ el índice de K^n después de todos los l -pasos. Por el Lema 4.3.8, los índices son los siguientes

$$\begin{aligned}
I_{i,-1}^n &= \pm 1 \\
I_{i,0}^n &= I_{i,-1}^n - \binom{n+1}{1} k_0 = \pm 1 - \binom{n+1}{1} k_0 \\
I_{i,1}^n &= I_{i,0}^n - \binom{n+1}{2} k_1 = \pm 1 - \binom{n+1}{1} k_0 - \binom{n+1}{2} k_1 \\
&\vdots \\
I_{i,n-1}^n &= I_{i,n-2}^n - \binom{n+1}{n} k_{n-1} = \pm 1 - \binom{n+1}{1} k_0 - \dots - \binom{n+1}{n} k_{n-1} \\
I_i^n &= \pm 1 - \sum_{j=0}^{n-1} \binom{n+1}{j+1} k_l
\end{aligned}$$

□

Ahora presentamos nuestro resultado principal.

Teorema 4.3.10 *Sea K^n una imagen dividida cromática, conexa y orientada coherentemente de σ^n , con estructura binaria simétrica y sin aristas de cruce. Entonces*

$$C(K^n) = \pm 1 - \sum_{l=0}^{n-1} \binom{n+1}{l+1} k_l$$

para algún $k_l \in \mathbb{Z}$.

Demostración: La demostración se obtiene del Lema 4.3.9 y de la Generalización del Lema de Índice 3.1.5.

□

Recuerde que en la sección 3.1.4 asumimos que K^n tiene estructura binaria simétrica bajo mapeos que preservan orden. De hecho, el proceso inductivo los desarrollamos bajo esta suposición. Sin embargo, el Teorema 4.3.10 también se cumple para imágenes divididas que cumplen la definición amplia de estructura binaria simétrica. La diferencia es que para este caso, un l -paso podría necesitar modificar mas de $\binom{n+1}{l+1}$ vértices para preservar la simetría de la coloración binaria. No obstante, este l -paso con más modificaciones simplemente representa varios l -pasos en el proceso que hemos mostrado.

4.4. Calculando el Valor k

En esta sección presentamos la demostración del Lema 4.3.3. Para esto, denotaremos como c^n a la coloración c (Definición 4.2.2) y como f^n a la permutación mínima f que utiliza. Considere una imagen dividida K^n de σ^n bajo ψ , como la

que se asume en el Teorema 4.3.10. Es decir, K^n es cromática, conexa, orientada coherentemente, con estructura binaria simétrica y sin aristas de cruce. Además, K^n tiene la coloración c^n . Agregamos una coloración más, muy parecida a c^n , a la imagen dividida de una $(n-1)$ -cara de σ^n .

Definición 4.4.1 Considere la cara σ_i^{n-1} de σ^n y el conjunto ID_i^{n-1} . La coloración c^n induce la coloración c_i^{n-1} a $\psi(\sigma_i^{n-1})$ definida como sigue

$$c_i^{n-1}(v) = \begin{cases} id(v) & \text{si } b(v) = 0 \\ f^{n-1}(id(v)) & \text{si } b(v) = 1 \end{cases}$$

donde

$$f^{n-1}(y) = \begin{cases} f^n(y) & \text{si } f^n(y) \neq i \\ f^n(i) & \text{si } f^n(y) = i \end{cases}$$

A continuación presentaremos algunas propiedades de c_i^{n-1} .

Lema 4.4.2 La función f^{n-1} es una permutación mínima de ID_i^{n-1} .

Demostración: Primero observe que ID_i^n es el dominio de f^{n-1} . Ya que f^n es una permutación de ID^n , existe un único $z \in ID_i^{n-1}$ tal que $f^n(z) = i$. Note que $f^{n-1}(z) = f^n(i) \in ID_i^{n-1}$. Sean I e I' subconjuntos de ID_i^{n-1} tales que $z \notin I'$ e $I = I' \cup \{z\}$. Por la definición de f^{n-1} , para todo $y \in I'$, tenemos que $f^{n-1}(y) = f^n(y)$. La restricción $f^{n-1}|_{I'}$ no es una permutación de I' , puesto que $f^n|_{I'}$ no es una permutación de I' . Ahora demostraremos que la condición se cumple para I . Supongamos, por contradicción, que la restricción $f^{n-1}|_I$ es una permutación de I . Supongamos que existe un único $y' \in I'$ tal que $f^{n-1}(y') \notin I'$. Advierta que si $f^{n-1}(y') = z$ y $f^{n-1}(z) \in I'$, entonces $f^{n-1}|_I$ es una permutación de I . De hecho, este es el único caso en que esto puede ser cierto. Puesto que $f^{n-1}(z) = f^n(i)$, tenemos que $f^n(i) \in I'$. Por lo tanto, $f^n|_{I \cup \{i\}}$ es una permutación de $I \cup \{i\}$. Una contradicción.

□

Observe que para toda imagen dividida de una cara σ_i^{n-1} de σ^n , c^n induce la coloración c_i^{n-1} a $\psi(\sigma_i^{n-1})$. Aún más, c_i^{n-1} induce este tipo de coloración a toda imagen dividida de una $(n-2)$ -cara de σ_i^{n-1} , y así sucesivamente. Para el resto de la sección, asumiremos que para toda cara σ^l de σ^n , $\psi(\sigma^l)$ tiene las coloraciones inducidas por todas aquellas $(l+1)$ -caras de σ^n que la contienen. Esto es, sean $\sigma_1^{l+1} \dots \sigma_m^{l+1}$ las $(l+1)$ -caras de σ^n , tales que $\sigma^l \subset \sigma_k^{l+1}$. La pseudo-variedad $\psi(\sigma^l)$ tiene la coloración c_j^l inducida por la coloración c^{l+1} de $\psi(\sigma_k^{l+1})$, donde j es el color del único vértice que está en σ_k^{l+1} , pero no en σ^l . Cuando no exista ambigüedad, simplemente escribiremos c^l .

La Figura 4.4 presenta una imagen dividida K^2 de un 2-simplejo σ^2 (a). Los vértices de K^2 tienen las coloraciones id , b y c^2 . La coloración c^2 utiliza la permutación mínima $f^2(x) = (x+1) \bmod 3$. La imagen dividida de la cara σ_2^1 (b) tiene la coloración c_2^1 , la cual utiliza la permutación mínima $f^1(x) = (x+1) \bmod 2$. Observe que c^2 y c_2^1 colorean el 1-simplejo marcado con

un círculo con los mismos colores.

Considere las caras $\sigma_1^0, \sigma_2^1 \dots \sigma_n^{n-1}$ de σ^n tales que $\sigma_1^0 \subset \sigma_2^1 \subset \dots \subset \sigma_n^{n-1}$. Asumiremos que $\psi(\sigma_i^{i-1})$ tiene la orientación inducida por $\psi(\sigma_{i+1}^i)$, $0 \leq i \leq n-1$, y $\psi(\sigma_n^{n-1})$ tiene la orientación inducida por K^n . Por el Lema 3.1.17, todas estas orientaciones son coherentes. Considere las caras σ_{l+1}^l y σ_{l+2}^{l+1} . También asumiremos que $\psi(\sigma_{l+2}^{l+1})$ tiene la coloración c^{l+1} y $\psi(\sigma_{l+1}^l)$ tiene la coloración inducida c_{l+1}^l por c^{l+1} . Denotaremos a $ID(\sigma_{l+2}^{l+1})$ como ID' . Por el resto de la sección, fijaremos las caras σ_{l+1}^l y σ_{l+2}^{l+1} .

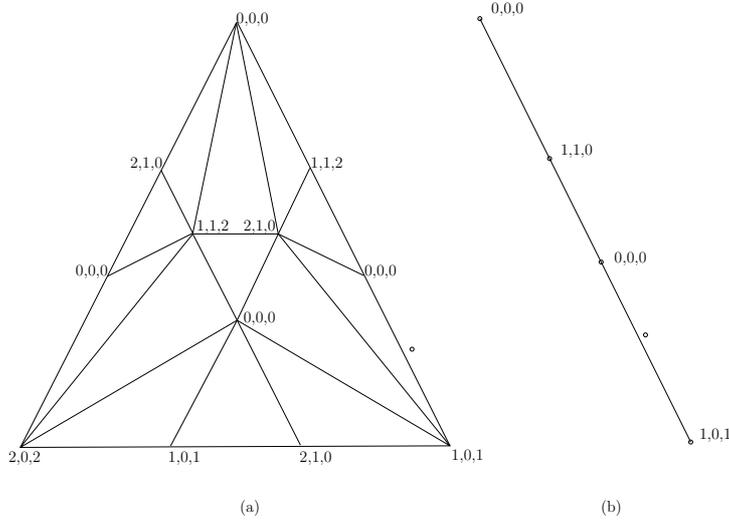


Figura 4.4: Una ejemplo de la coloración inducida por c^n .

Lema 4.4.3 *Un l -simplejo de $\psi(\sigma_{l+1}^l)$ que no es 1-monocromático, está coloreado propiamente bajo c^{l+1} con ID'_{l+1} si y solo si está coloreado propiamente bajo c_{l+1}^l .*

Demostración: Sea τ un l -simplejo de $\psi(\sigma_{l+1}^l)$ tal que no es 1-monocromático. Por el Lema 4.2.4, $c^{l+1}(\tau) = ID'_{l+1}$ si y solo si τ es 0-monocromático, y por el Lema 4.2.3, τ está coloreado propiamente bajo c_{l+1}^l si y solo si es monocromático. Para probar la otra dirección, advierta que si τ está coloreado propiamente bajo c_{l+1}^l , entonces es monocromático. Puesto que τ^l no es 1-monocromático, entonces $c^{l+1}(\tau) = ID'_{l+1}$.

□

Del lema anterior se desprende el siguiente corolario.

Corolario 4.4.4 *Sea v un vértice de $\psi(\sigma_{i+1}^l)$ tal que $b(v) = 0$. Un l -simplejo de $\psi(\sigma_{i+1}^l)$ está coloreado propiamente bajo c^{l+1} con ID'_{i+1} si y solo si está coloreado propiamente bajo c_{i+1}^l .*

Demostración: Observe que todo l -simplejo de $st(v, \psi(\sigma_{i+1}^l))$ no es 1-monocromático. El resto de la prueba se obtiene del Lema 4.4.3.

□

Ahora presentamos la forma en que conectamos la idea anterior con la Generalización del Lema del Índice. El siguiente lema asume que el valor k de $st(v, bd(\psi(\sigma_{i+2}^{l+1})))$ es calculado con respecto a c^{l+1} , mientras que el contenido de $st(v, \psi(\sigma_{i+1}^l))$ es calculado con respecto a c_{i+1}^l . De igual forma que cuando calculamos el valor k de $st(v, L)$, simplemente escribiremos $C(v, L)$ cuando calculemos el contenido de $st(v, L)$.

Lema 4.4.5 *Sea v un vértice de $\psi(\sigma_{i+1}^l)$ tal que tiene un carrier de dimensión l' y $b(v) = 0$. Si para todo vértice interno $u \in \psi(\sigma)$, $b(u) = 0$, donde $\sigma \subset \sigma_{i+2}^{l+1}$ y $l' < \dim(\sigma) < l + 1$, entonces*

$$k^m(v, bd(\psi(\sigma_{i+2}^{l+1})), ID'_{i+1}) = C(v, \psi(\sigma_{i+1}^l))$$

Demostración: Primero recuerde que tanto $k^m(v, bd(\psi(\sigma_{i+2}^{l+1})), ID'_{i+1})$ como $C(v, \psi(\sigma_{i+1}^l))$ cuentan los l -simplejos de $st(v, bd(\psi(\sigma_{i+2}^{l+1})))$ y $st(v, \psi(\sigma_{i+1}^l))$, respectivamente, que están coloreados propiamente con ID'_{i+1} . Por el Lema, 3.1.8 (1), $st(v, \psi(\sigma_{i+1}^l)) \subseteq st(bd(\psi(\sigma_{i+2}^{l+1})))$. Utilizando un argumento similar al que se ocupó en la demostración del Lema 4.3.7, podemos probar que $st(v, \psi(\sigma_{i+1}^l))$ contiene todos los l -simplejos de $st(v, bd(\psi(\sigma_{i+2}^{l+1})))$ que están coloreados propiamente bajo c^{l+1} con ID'_{i+1} . Por el Corolario 4.4.4, podemos contar estos simplejos con c_{i+1}^l . Por lo tanto,

$$k^m(v, bd(\psi(\sigma_{i+2}^{l+1})), ID'_{i+1}) = C(v, \psi(\sigma_{i+1}^l))$$

donde el valor k de $st(v, bd(\psi(\sigma_{i+2}^{l+1})))$ es calculado con respecto a c^{l+1} , mientras que el contenido de $st(v, \psi(\sigma_{i+1}^l))$ es calculado con respecto a c_{i+1}^l .

□

De manera informal, el siguiente lema dice que cuando los vértices internos de $\psi(\sigma_{i+2}^{l+1})$ tienen color binario 0, los l -simplejos de $lk(v, \psi(\sigma_{i+2}^{l+1}))$ que están coloreados propiamente con $ID(\sigma_{i+1}^l)$ están concentrados en $\psi(\sigma_{i+1}^l)$.

Lema 4.4.6 *Sea v un vértice de $\psi(\sigma_{i+1}^l)$ tal que tiene un carrier de dimensión l' . Considere un simplejo $\rho^l \in lk(v, \psi(\sigma_{i+2}^{l+1}))$ tal que $c^{l+1}(\rho^l) = ID'_{i+1}$. Si para todo vértice interno $u \in \psi(\sigma)$, $b(u) = 0$, donde $\sigma \subset \sigma_{i+2}^{l+1}$ y $l' < \dim(\sigma) < l + 1$, entonces $\rho^l \in st(v, \psi(\sigma_{i+1}^l))$.*

Demostración: Por la Afirmación 3.1.10, v es un vértice externo de $\psi(\sigma_{l+2}^{l+1})$. Además, por el Lema 3.1.8 (1), $st(v, \psi(\sigma_{l+1}^l)) \subset lk(v, \psi(\sigma_{l+2}^{l+1}))$. Supongamos, por contradicción, que $\rho^l \notin st(v, \psi(\sigma_{l+1}^l))$. Tenemos los siguientes dos casos.

Para el primer de ellos tenemos que existe una cara σ^l de σ_{l+2}^{l+1} tal que $\rho^l \in \psi(\sigma^l)$. Observe que $ID(\sigma^l) \neq ID'_{l+1}$. Aún más, $l+1 \in ID(\sigma^l)$. Es decir, existe $w \in \rho^l$ tal que $id(w) = l+1$. Ahora considere la cara $\sigma^{l'}$ de σ_{l+2}^{l+1} tal que $v \in \psi(\sigma^{l'})$. Advierta que $l+1 \notin ID(\sigma^{l'})$ y, por tanto, $w \notin \psi(\sigma^{l'})$. Todavía más, note que si w tiene un carrier de dimensión l' , entonces K^n tiene una arista de cruce. Por lo tanto, w tiene un carrier de dimensión mayor que l' . Es decir, $b(w) = 0$. Por ende, $c(w) = l+1$ y $c(\rho^l) \neq ID'_{l+1}$. Una contradicción.

Para el segundo caso tenemos que no existe una cara σ^l de σ_{l+2}^{l+1} tal que $\rho^l \in \psi(\sigma^l)$. Primero observe que $v \notin \rho^l$, ya que $\rho^l \in lk(v, \psi(\sigma_{l+2}^{l+1}))$. Por lo tanto, existe $w \in \rho^l$ tal que $id(w) = l+1$. Note que $w \notin \psi(\sigma_{l+1}^l)$. De forma similar que en el caso anterior, w tiene un carrier de dimensión mayor que l' . Por ende, $c(w) = l+1$ y $c(\rho^l) \neq ID'_{l+1}$. Una contradicción. Esto completa la demostración. \square

Ahora, presentamos la demostración del Lema 4.3.3.

Lemma 4.3.3 *Considere la cara σ_{l+1}^l . Sea v un vértice de $\psi(\sigma_{l+1}^l)$ tal que $b(v) = 0$. Si para $l < p < n$, para todo vértice interno $u \in \psi(\sigma_{p+1}^p)$, $b(u) = 0$, entonces*

$$k^{n-1}(v, bd(K^n), ID_n^n) = (-1)^e k^l(v, \psi(\sigma_{l+1}^l), ID_I^n)$$

donde $e = ((l+1) + (l+2) + \dots + (n-1))$ e $I = \{l+1, l+2, \dots, n\}$.

Demostración: Primero observe que $v \in \psi(\sigma_{m+1}^m)$, $l \leq m \leq n-1$. Además, para todo vértice interno $u \in \psi(\sigma_{p+1}^p)$, $b(u) = 0$, donde $l+1 \leq p \leq n-1$. Ahora, por el Lema 4.4.5

$$k^{n-1}(v, bd(K^n), ID_n^n) = C(v, \psi(\sigma_n^{n-1}))$$

donde el contenido de $st(v, \psi(\sigma_n^{n-1}))$ es calculado con respecto a c_n^{n-1} , y por la Generalización del Lema del Índice 3.1.5, tenemos que $C(v, \psi(\sigma_n^{n-1})) = (-1)^{n-1} I_{n-1}(v, \psi(\sigma_n^{n-1}))$. Todavía más, por el Lema 4.4.6, $st(v, \psi(\sigma_{n-1}^{n-2}))$ contiene todo los $(n-2)$ -simplejos de $lk(v, \psi(\sigma_n^{n-1}))$ que están coloreados propiamente bajo c_n^{n-1} con $ID_{\{n-1, n\}}^n$. Recuerde que la coloración c_n^{n-1} de $\psi(\sigma_n^{n-1})$ induce la coloración c_{n-1}^{n-2} a $\psi(\sigma_{n-1}^{n-2})$. Por el Lema 4.4.4, un $(n-2)$ -simplejo de $\psi(\sigma_n^{n-1})$ está coloreado propiamente bajo c_{n-1}^{n-2} si y solo si lo está bajo c_n^{n-1} con $ID_{\{n-1, n\}}^n$. Esto es, podemos contar estos simplejos con c_{n-1}^{n-2} . Por el Lema 3.1.6, sabemos que $I_{n-1}(v, \psi(\sigma_n^{n-1})) = C(v, \psi(\sigma_{n-1}^{n-2}))$, donde el contenido de $st(v, \psi(\sigma_{n-1}^{n-2}))$ es calculado con respecto a c_{n-1}^{n-2} . Es decir, $C(v, \psi(\sigma_n^{n-1})) = (-1)^{n-1} C(v, \psi(\sigma_{n-1}^{n-2}))$. Advierta que podemos repetir lo anterior con $\psi(\sigma_{n-1}^{n-2})$ y $\psi(\sigma_{n-2}^{n-3})$. En general, para las caras σ_{q+1}^q y σ_{q+2}^{q+1} , $l \leq q \leq n-2$, $C(v, \psi(\sigma_{q+2}^{q+1})) =$

$(-1)^{q+1}C(v, \psi(\sigma_{q+1}^q))$, donde el contenido de $st(v, \psi(\sigma_{q+2}^{q+1}))$ es calculado con respecto a c^{q+1} y el contenido de $st(v, \psi(\sigma_{q+1}^q))$ con respecto a c_{q+1}^q . Por lo tanto

$$C(v, \psi(\sigma_n^{n-1})) = (-1)^{((l+1)+(l+2)+\dots+(n-1))}C(v, \psi(\sigma_{l+1}^l))$$

Ahora considere un simplejo $\tau \in st(v, \psi(\sigma_{l+1}^l))$. Por el Lema 4.2.4, $c^n(\tau) = id(\tau)$ si y solo si τ es 0-monocromático, y por el Lema 4.2.3, $c_{l+1}^l(\tau) = id(\tau)$ si y solo si τ es monocromático. Note que τ no es 1-monocromático, ya que $b(v) = 0$. Por lo tanto, podemos contar los l -simplejos de $st(v, \psi(\sigma_{l+1}^l))$ que están coloreados propiamente bajo c_{l+1}^l , con c^n . Observe que $id(\tau) = ID_{\{l+1, l+2, \dots, n\}}^n$. Al final tenemos que

$$C(v, \psi(\sigma_{l+1}^l)) = k^l(v, \psi(\sigma_{l+1}^l), ID_{\{l+1, l+2, \dots, n\}}^n)$$

□

4.5. Cuando el Índice es Igual a Cero

Ayudados de la caracterización que presentamos en la sección 4.3, a continuación construimos una imagen dividida de dimensión 5, como la que el Teorema 4.3.10 asume, tal que su índice con respecto a c es igual a cero.

Sea \mathbb{K}^5 una imagen dividida de σ^5 bajo ψ , como las que asume el Teorema 4.3.10. Además, \mathbb{K}^5 tiene la coloración c , la cual usa la permutación mínima $f(x) = (x+1) \bmod 6$. Fijaremos \mathbb{K}^5 por el resto de la sección.

4.5.1. Estructura de \mathbb{K}^5

Primero describiremos la estructura de \mathbb{K}^5 , para después, en la siguiente sección, demostrar que su índice es igual a cero. En un principio asumiremos que todo vértice de \mathbb{K}^5 tiene color binario 0, a menos que se especifique lo contrario.

Para toda cara σ^0 de σ^5 , $\psi(\sigma^0)$ es un vértice v con $id(v) = ID(\sigma^0)$ y $b(v) = 1$. Además, para toda cara σ^1 de σ^5 , $\psi(\sigma^1)$ es un 1-camino con 7 1-simplejos coloreados propiamente con $ID(\sigma^1)$. Aún más, el 1-simplejo en el centro de $\psi(\sigma^1)$ es 1-monocromático. En la Figura 4.5 (c) se puede ver un ejemplo $\psi(\sigma^1)$, donde los vértices tienen asociadas las coloraciones id , b y c , en ese orden. Observe que los vértices en la frontera tienen color binario 1. Estos dos son imágenes divididas de dos 0-caras de σ^5 . Es fácil ver que $\psi(\sigma^1)$ es una pseudovariedad orientable, cromática y conexa.

Ahora considere una cara σ^l de σ^5 , $2 \leq l \leq 5$. El complejo $\psi(\sigma^l)$ contiene un solo l -simplejo τ , tal que todos sus vértices son vértices internos de $\psi(\sigma^l)$. Por supuesto que $id(\tau) = ID(\sigma^l)$ y, además, para el vértice $v \in \tau$ que tiene el id más grande de entre los vértices de τ , $b(v) = 1$. Intuitivamente, $\psi(\sigma^l)$ es el cono que se forma uniendo τ a $bd(\psi(\sigma^l))$. Los otros simplejos de $\psi(\sigma^l)$ se definen de la siguiente manera. Para $l' < l$, sea $\sigma^{l'}$ una cara de σ^l . Considere un l' -simplejo $\rho \in \psi(\sigma^{l'})$. El complejo $\psi(\sigma^l)$ contiene el l -simplejo $\rho \cup \gamma_2$, donde

$\gamma_2 \subset \tau$ e $id(\gamma_2) = ID(\sigma^l) - ID(\sigma^{l'})$. Observe que $id(\rho \cup \gamma_2) = ID(\sigma^l)$. De manera informal, γ_2 completa los colores de ρ . La Figura 4.7 muestra un ejemplo de $\psi(\sigma^2)$ en el que los vértices están etiquetados con b .

Por construcción, $\psi(\sigma^l)$ es una pseudovariiedad cromática, orientable y conexa, y, por tanto, \mathbb{K}^5 es una imagen dividida cromática, orientable y conexa de σ^5 . Advierta que también tiene estructura binaria simétrica y no tiene aristas de cruce. Aún más, todo 5-simplejo de \mathbb{K}^5 contiene por lo menos un vértice interno de \mathbb{K}^5 .

Ahora asumiremos que \mathbb{K}^5 tiene una orientación coherente, es decir, la pseudovariiedad $\mathbb{K}^5 = \psi(\sigma^5)$ tiene una orientación coherente. La característica principal de \mathbb{K}^5 es que su índice con respecto a c es igual a cero. Esto será de gran importancia en el capítulo 5.

4.5.2. Índice Igual a Cero

Ahora probaremos que el índice de \mathbb{K}^5 con respecto a c es cero, para lo cual utilizaremos el proceso inductivo descrito en la sección 4.3. Primero observe que por el Lema 4.3.9, el índice de \mathbb{K}^5 con respecto a c tiene la forma

$$\begin{aligned} I_5^5 &= \pm 1 - \sum_{l=0}^4 \binom{6}{l+1} k_l \\ I_5^5 &= \pm 1 - \binom{6}{1} k_0 - \binom{6}{2} k_1 - \binom{6}{3} k_2 - \binom{6}{4} k_3 - \binom{6}{5} k_4 \\ I_5^5 &= \pm 1 - 6k_0 - 15k_1 - 20k_2 - 15k_3 - 6k_4 \end{aligned}$$

para algunos $k_0, k_1, k_2, k_3, k_4 \in \mathbb{Z}$.

Asumiremos que \mathbb{K}^5 tiene una orientación tal que $I_5^5 = 1$ al comienzo del proceso. Por lo tanto, el índice es

$$I_5^5 = 1 - 6k_0 - 15k_1 - 20k_2 - 15k_3 - 6k_4$$

Advierta que si $k_0 = 1, k_1 = 1, k_2 = -1, k_3 = 0$ y $k_4 = 0$, entonces $I_5^5 = 0$. Además, recuerde que los l -pasos en el proceso son representados por k_l . Demostraremos que el proceso inductivo aplicado a \mathbb{K}^5 da estos valores.

Considere las caras $\sigma_1^0, \sigma_2^1, \sigma_3^2, \sigma_4^3, \sigma_5^4$ de σ^5 tales que $\sigma_1^0 \subset \sigma_2^1 \subset \sigma_3^2 \subset \sigma_4^3 \subset \sigma_5^4$. Para $0 \leq l \leq 3$, asumiremos que $\psi(\sigma_{l+1}^l)$ tiene la orientación inducida por $\psi(\sigma_{l+2}^{l+1})$ y, además, asumiremos que $\psi(\sigma_5^4)$ tiene la orientación inducida por \mathbb{K}^5 .

$k_0 = 1$. Recuerde que en este punto del proceso, para todo vértice $u \in bd(\mathbb{K}^5)$, $b(u) = 0$. Considere el vértice $v \in \psi(\sigma_1^0)$. Note que v tiene un carrier de dimensión 0 y $c(v) = 0$. Por el Lema 4.3.8

$$k_0 = k^4(v, bd(\mathbb{K}^5), ID_5^5)$$

Además, por el Lema 4.3.3,

$$k^4(v, bd(\mathbb{K}^5), ID_5^5) = (-1)^{(1+2+3+4)} k^0(v, \psi(\sigma_1^0), ID_{\{1+2+3+4+5\}}^n)$$

De la demostración del Lema 4.3.1, podemos ver que $k^0(v, \psi(\sigma_1^0), ID_I^n) = C(\psi(\sigma_1^0))$, donde $I = \{1+2+3+4+5\}$. Además, $C(\psi(\sigma_1^0)) = 1$, ya que $I_5^5 = 1$. Por lo tanto, $k^4(v, bd(\mathbb{K}^5), ID_5^5) = 1$.

$k_1 = 1$. En el caso anterior, $st(v, \psi(\sigma_1^0))$ contiene únicamente un 0-simplejo ρ^0 , el cual es contado como $+1$ por $k^0(v, \psi(\sigma_1^0), ID_{\{1+2+3+4+5\}}^5)$. Por el Lema 4.2.6, ρ^0 tiene orientación $+1$. Todavía más, ρ^0 es cara de un 1-simplejo τ^1 de 1-esquinas($\psi(\sigma_2^1)$). Observe que la orientación de ρ^0 es la orientación de τ^1 multiplicada por $(-1)^1$. Por ende, τ^1 tiene la orientación -1 . Por el Lema 3.1.21, todos los simplejos en 1-esquinas($\psi(\sigma_2^1)$) tienen orientación -1 .

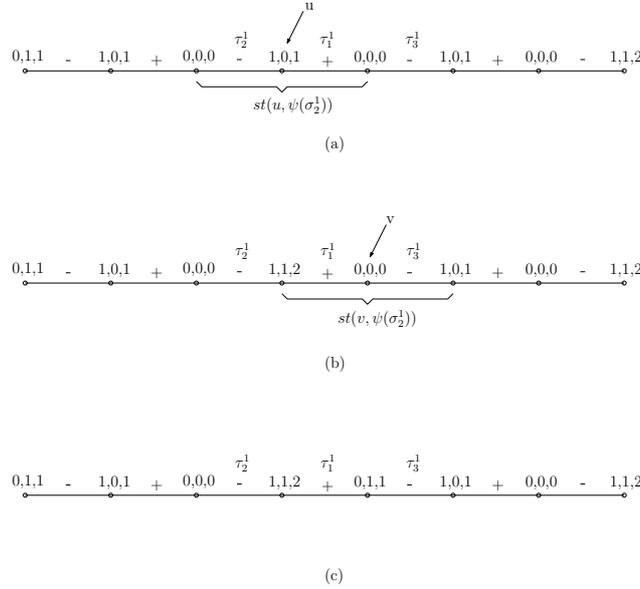


Figura 4.5: Dos 1-pasos sobre \mathbb{K}^5 .

La Figura 4.5 muestra los dos 1-pasos a realizar sobre \mathbb{K}^5 , en los que los vértices tienen las coloraciones id , b y c , en este orden. Puesto que únicamente un 0-paso se ha realizado, los dos vértices en la frontera tienen su coloración binaria en 1. Considere los 1-simplejos τ_1 , τ_2 y τ_3 que se ven en la figura. Por el Lema 4.3.8

$$k_1 = k^4(u, v, bd(\mathbb{K}^5), ID_5^5)$$

donde u y v son los vértices de τ_1 . Además, por el Corolario 4.3.4

$$k^4(u, v, bd(\mathbb{K}^5), ID_5^5) = (-1)^{(2+3+4)} k^1(u, v, \psi(\sigma_2^1), ID_{\{2+3+4+5\}}^5)$$

Advierta que $st(u, \psi(\sigma_2^1)) \cup st(v, \psi(\sigma_2^1))$ contiene a τ_1 , τ_2 y τ_3 , junto con todas sus caras. Tenemos que $k^1(u, v, \psi(\sigma_2^1), ID_{\{2+3+4+5\}}^5) = -1$ y, por tanto,

$$k^4(u, v, bd(\mathbb{K}^5), ID_5^5) = 1.$$

$\underline{k_2 = -1}$. De forma similar que en el caso anterior, observe que todo simplejo ρ^1 de 1 -esquinas($\psi(\sigma_2^1)$) es cara de un simplejo τ^2 de 2 -esquinas($\psi(\sigma_3^2)$). Además, la orientación de ρ^1 es la orientación de τ^2 multiplicada por $(-1)^2$. Ya que los simplejos de 1 -esquinas($\psi(\sigma_2^1)$) tienen orientación -1 , la orientación de todos los simplejos de 2 -esquinas($\psi(\sigma_3^2)$) es -1 .

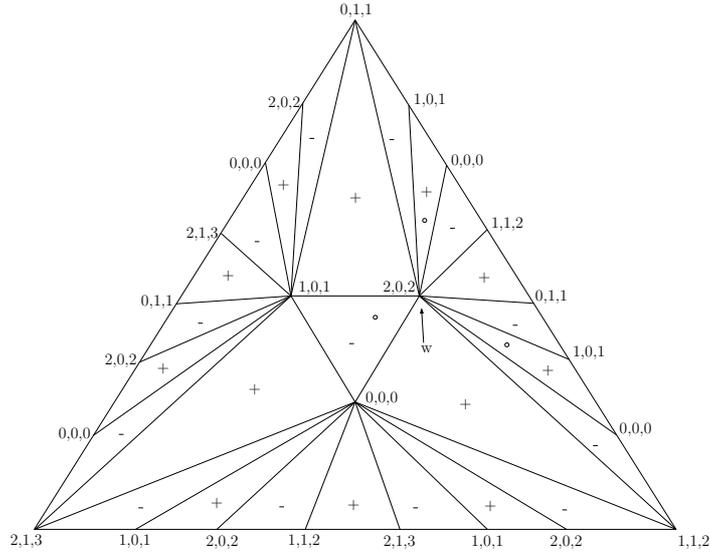


Figura 4.6: Un 2-paso sobre \mathbb{K}^5 .

La Figura 4.6 muestra el 2-paso a realizar sobre \mathbb{K}^5 , en el cual los vértices tienen las coloraciones id , b y c . Observe que algunos vértices en la frontera tienen color binario 1. Estos son los vértices procesados en los pasos anteriores. El complejo $st(w, \psi(\sigma_3^2))$ contiene a los 2-simplejos, junto con todas sus caras, alrededor de w . Los 2-simplejos de $st(w, \psi(\sigma_3^2))$ coloreados propiamente bajo c con $ID_{\{3+4+5\}}^5$ están marcados con un círculo. Por el Lema 4.3.8

$$k_2 = k^4(w, bd(\mathbb{K}^5), ID_5^5)$$

Todavía más, por el Corolario 4.3.3

$$k^4(w, bd(\mathbb{K}^5), ID_5^5) = (-1)^{(3+4)} k^2(w, \psi(\sigma_3^2), ID_{\{3+4+5\}}^5)$$

Observe que $k^2(w, \psi(\sigma_3^2), ID_{\{3+4+5\}}^5) = 1$. Por lo tanto, $k^4(w, bd(\mathbb{K}^5), ID_5^5) = -1$.

$\underline{k_3 = 0}$ y $\underline{k_4 = 0}$. Ya que todos los demás vértices de \mathbb{K}^5 tienen color binario 0, el proceso no realiza 3-pasos ni 4-pasos. Por lo tanto, $k_3 = 0$ y $k_4 = 0$.

Al final del proceso tenemos que $I_5^5 = 0$. Además, por la Generalización del Lema del Índice 3.1.5, $C^5 = 0$. Podemos verificar todo esto muy fácilmente.

Sea τ el 5-simplejo de \mathbb{K}^5 que contiene todos los vértices internos de \mathbb{K}^5 . Supongamos que este es 1-monocromático. Puesto que, por construcción, todo 5-simplejo contiene por lo menos un vértice de τ , \mathbb{K}^5 no contiene 5-simplejos 0-monocromáticos. De la siguiente manera podemos verificar que el índice de \mathbb{K}^5 es cero. Considere una cara σ^l de σ^5 , $0 \leq l \leq 4$. Advierta que todo l -simplejo 1-monocromático de $\psi(\sigma^l)$ genera un 5-simplejo 1-monocromático. Por construcción, si $0 \leq l \leq 2$, entonces $\psi(\sigma^l)$ contiene únicamente un l -simplejo 1-monocromático, y en cualquier otro caso, $\psi(\sigma^l)$ no contiene l -simplejos de este tipo. Es decir, toda cara de σ^5 de dimensión menor o igual a 2 genera un 5-simplejo 1-monocromático y, por tanto, \mathbb{K}^5 tiene $1 + \binom{6}{1} + \binom{6}{2} + \binom{6}{3}$ 5-simplejos 1-monocromáticos.

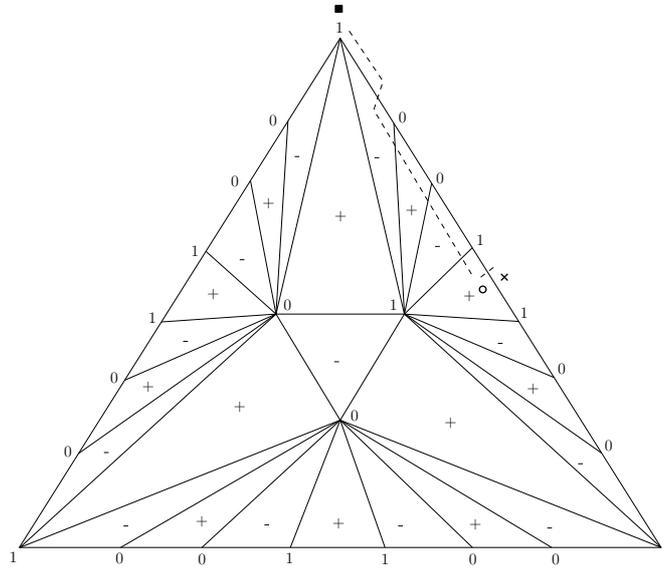


Figura 4.7: Dos caminos entre simplejos monocromáticos de \mathbb{K}^5 .

Ahora bien, por el Lema 4.2.6, C^5 cuenta los 5-simplejos 1-monocromáticos de \mathbb{K}^5 por orientación, multiplicándolas por -1 . Sea S^l el conjunto que consiste en los 5-simplejos 1-monocromáticos de \mathbb{K}^5 generados por las imágenes divididas de las l -caras de σ^5 . Observe que la única forma en que $C^5 = 0$, es que los simplejos en $A = S^2 \cup \{\tau\}$ tengan orientación d y los que están en $B = S^0 \cup S^1$ tengan orientación $-d$. Por lo tanto, por el Lema 3.1.1, podemos afirmar que todo 5-camino en \mathbb{K}^5 de un simplejo en A a otro en B , tiene un número par de 5-simplejos. Esto no es difícil de verificar.

La Figura 4.7 muestra la imagen dividida de una 2-cara de σ^5 . Los vértices únicamente tiene asociada la coloración b . Sean τ_{circ} , τ_{cruz} y τ_{cuad} los 5-simplejos 1-monocromáticos generados por el triángulo, arista y vértice marcados con un

circulo, cruz y cuadro, respectivamente. Advierta que τ_{circ} y τ_{cruz} comparten una 4-cara. Por lo tanto, por el Lema 3.1.1, estos dos simplejos tienen orientaciones opuestas. Ahora considere los simplejos cubiertos por la línea punteada que va del triángulo marcado con el círculo, al vértice marcado con el cuadro. Estos simplejos son cuatro triángulos, una arista y un vértice. Los 5-simplejos generados por estos simplejos forma un 5-camino de τ_{circ} a τ_{cuad} . Por lo tanto, τ_{circ} y τ_{cuad} tienen orientaciones opuestas. Aún más, τ_{cuad} y τ comparten una 4-cara y, por tanto, tienen orientaciones opuestas también. Observe que τ y τ_{circ} tienen la misma orientación. Podemos revisar uno a uno los caminos entre los simplejos de A y B y comprobar que todos ellos tienen un número par de simplejos y, por ende, $C^5 = 0$. Finalmente, por la Generalización del Lema del Índice 3.1.5, $-I_5^5 = 0$.

Capítulo 5

Consecuencias y Conclusiones

En este capítulo presentamos las implicaciones de los resultados obtenidos en el capítulo 4. En la sección 5.1 mostramos la primera demostración combinatoria de imposibilidad del k -renombrado, $k < 2n$, para una infinidad de casos, así como las implicaciones del complejo \mathbb{K}^5 , construido en la sección 4.5, hacia las otras dos demostraciones de imposibilidad del renombrado. Asimismo, nuestras conclusiones y el trabajo futuro aparecen en las secciones 5.2 y 5.3, respectivamente.

5.1. Consecuencias

En esta sección presentamos las implicaciones del Teorema 4.3.10 y del complejo \mathbb{K}^5 hacia el problema del renombrado.

5.1.1. Demostración Combinatoria de Imposibilidad del Renombrado

Como se muestra en la sección 3.2.1, la tarea del renombrado se modela como la tripleta $\langle I^n, O^n, \Delta \rangle$. El complejo de entrada I^n es el complejo que contiene todos los n -simplejos, junto con todas sus caras, formados por los subconjuntos de tamaño $n + 1$ del dominio de nombres originales I . De forma similar se define el complejo de salida O^n sobre el dominio de nombres nuevos O . El mapeo Δ mapea todo n -simplejo de I^n a todos los n -simplejos de O^n .

Sea A un algoritmo que soluciona el k -renombrado. El algoritmo A implica la existencia de una imagen dividida K^n cromática, orientable y conexa de $\sigma^n \in I^n$, con estructura binaria simétrica y sin aristas de cruce, tal como lo expone la sección 3.2.5. De hecho, K^n representa todas las ejecuciones en las que A comienza con las entradas en σ^n . Además, la coloración binaria b de K^n representa las paridades de los nombres nuevos que deciden los procesadores.

Observe que K^n es una imagen dividida como la que el Teorema 4.3.10 asume. Por lo tanto, este implica una caracterización combinatoria del número de ejecuciones en las cuales A renombra los procesadores haciendo uso únicamente de nombres pares o impares, respectivamente, de O . Por otro lado, advierta que si $k < 2n$, entonces el dominio de nombres nuevos O no tiene $n+1$ nombres pares o impares y, por ende, K^n no puede tener n -simplejos monocromáticos cuando A utiliza este rango de nombres nuevos. De lo anterior podemos concluir que si el contenido de K^n con respecto a c es distinto de cero, entonces por lo menos existe una ejecución en la cual dos procesadores eligen el mismo nombre nuevo.

Corolario 5.1.1 *Si no existen $k_l \in \mathbb{Z}$, $0 \leq l \leq n-1$, tales que la ecuación del Teorema 4.3.10 se hace cero, $C^n = 0$, entonces no existe un algoritmo anónimo libre de espera que solucione el k -renombrado, $k < 2n$.*

De forma más precisa, lo que buscamos saber es si existen $k_0, k_1 \dots k_{n-1}$ tales que la ecuación

$$\binom{n+1}{1}k_0 + \binom{n+1}{2}k_2 + \dots + \binom{n+1}{n}k_{n-1} = \pm 1$$

tiene solución. Para que esto suceda, los coeficientes $\binom{n+1}{1}, \binom{n+1}{2} \dots \binom{n+1}{n}$ deben ser primos relativos [12, Sección 2.5]. Esto nos da el siguiente resultado, el cual es un caso especial del Teorema 6.2 en [19] y el Teorema 6.13 en [18].

Corolario 5.1.2 *Si $\binom{n+1}{1}, \binom{n+1}{2}, \dots, \binom{n+1}{n}$ no son primos relativos, entonces no existe un algoritmo anónimo libre de espera que solucione el k -renombrado, $k < 2n$.*

El siguiente corolario muestra un infinidad de casos para los cuales no existe el algoritmo de renombrado del que habla el corolario anterior.

Corolario 5.1.3 *Si $n+1$ es primo o $n=3$, entonces no existe un algoritmo anónimo libre de espera que solucione el k -renombrado, $k < 2n$.*

Demostración: Primero veremos el caso $n=3$. Tenemos que los coeficientes $\binom{4}{1} = 4$, $\binom{4}{2} = 6$ y $\binom{4}{3} = 4$ son divisibles por 2 y por tanto no son primos relativos.

Ahora considere el caso en que $n+1$ es primo. Advierta que $n+1$ es factor de $\binom{n+1}{i+1}$, $0 \leq i \leq n-1$, y, por tanto, $\binom{n+1}{1}, \binom{n+1}{2} \dots \binom{n+1}{n}$ son divisibles por $n+1$.

□

En otras palabras, lo que demuestra el Corolario 5.1.3 es que para todos estos casos las características topológicas del complejo de las ejecuciones de un algoritmo de renombrado impiden que exista el mapeo simplicial del que se habla en la Afirmación 3.2.1 y el Teorema 3.2.2.

5.1.2. Implicaciones de \mathbb{K}^5

Considere la imagen dividida \mathbb{K}^5 construida en la sección 4.5. Asuma que esta tiene una orientación coherente. Supongamos que para todo vértice interno v de \mathbb{K}^5 , $b(v) = 0$. Puesto que todo 5-simplejo contiene por lo menos uno de estos vértices, ningún simplejo de esta dimensión es 1-monocromático y, por tanto, por el Lema 4.2.6, el contenido de \mathbb{K}^5 con respecto a c cuenta los 5-simplejos por orientación. Esto implica directamente que el Teorema 6.2 en [3] es falso.

Teorema 5.1.4 ([3, Teorema 6.2]) *Asuma que K^n es una imagen dividida cromática de σ^n bajo ψ , con estructura binaria simétrica. Si K^n es orientable y conexa, entonces el número de los n -simplejos monocromáticos de K^n contados por orientación es distinto de cero.*

La demostración del teorema anterior ocupa el Lema 6.1 en [19]. La pregunta es, ¿el Lema 6.1 en [19] es falso, o la conexión entre éste y el Teorema 6.2 en [3] es incorrecta?

El apéndice B en [3] estudia la relación que existe entre las imágenes divididas y los mapeos entre cadenas de simplejos o *chain maps*¹. El resultado de este estudio es que existe una relación estrecha entre los mapeos de cadenas y las imágenes divididas: una imagen dividida induce un mapeo de cadenas. De hecho, la coloración c de la imagen dividida \mathbb{K}^5 también induce un mapeo de cadenas.

Por otro lado, que el índice de \mathbb{K}^5 sea cero, intuitivamente quiere decir que la frontera de \mathbb{K}^5 se envuelve cero veces en la frontera de σ^5 . Aún más, podemos pensar el mapeo ψ y la Generalización del Lema del Índice, de forma conjunta, como una composición ϕ que resulta en un mapeo de σ^5 a σ^5 tal que la frontera del simplejo se mapea cero veces a ella misma (recuerde que ψ mapea la frontera de σ^5 a la frontera de \mathbb{K}^5). Esto implica que el Lema 6.1 en [19] es falso, ya que este afirma que un mapeo con las características de ϕ siempre envuelve un número distinto de cero veces la frontera de σ^5 a ella misma. Por lo tanto, la demostración de imposibilidad del k -renombrado en [19], Teorema 6.2, es incorrecta, ya que el argumento principal de la prueba se basa en el Lema 6.1.

Asimismo, \mathbb{K}^5 también implica que la demostración de imposibilidad del renombrado en [18] es incorrecta. Lo que realmente se demuestra allí es que no existe un algoritmo libre de espera que solucione la versión reducida del renombrado, en la cual los procesadores tienen que decidir 0 o 1. La condición es que en toda ejecución por lo menos un procesador debe decidir 0 y por lo menos uno debe decidir 1. Esto implica que tampoco existe un algoritmo libre de espera que solucione el renombrado. El argumento principal de la demostración es que la frontera de una pseudovariiedad con las características de \mathbb{K}^5 se mapea necesariamente un número diferente de cero veces alrededor de la frontera de un simplejo específico del complejo de salida. Esta frontera realmente es un hoyo en

¹Una cadena de simplejos es una suma formal de simplejos de igual dimensión, cada uno multiplicado por un coeficiente entero

el complejo. Sin embargo, como ya se menciona antes, \mathbb{K}^5 demuestra que existe un complejo con las características deseadas y que mapea su frontera cero veces alrededor de ese hoyo.

5.2. Conclusiones

A lo largo de la tesis estudiamos el complejo de un algoritmo de renombrado libre de espera en el modelo asíncrono de memoria compartida, para lo cual utilizamos herramientas de topología combinatoria tales como son las imágenes divididas y la Generalización del Lema del Índice. El resultado de este estudio es una caracterización combinatoria del número de ejecuciones en las cuales el algoritmo renombra los procesadores haciendo uso única y exclusivamente de nombres pares o impares, respectivamente, del dominio de nombres nuevos (Teorema 4.3.10).

Como se explicó en la sección anterior, esta caracterización implica la primera demostración totalmente combinatoria, para una infinidad de casos, de imposibilidad del k -renombrado, $k < 2n$ (Corolario 5.1.3). Todavía más, para determinar si para una n en específico no existe un algoritmo como éste, es suficiente verificar que los coeficientes binomiales $\binom{n+1}{1}, \binom{n+1}{2} \dots \binom{n+1}{n}$ no son primos relativos (Corolario 5.1.2). Cabe señalar que esto se puede hacer fácil y rápidamente con el famoso algoritmo de Euclides para calcular el máximo común divisor [12, Sección 2.6]. Además, ayudados de nuestra caracterización, construimos una imagen dividida de dimensión 5 que prueba que las otras dos demostraciones existentes tienen un error. Por lo tanto, la nuestra es la primera demostración de imposibilidad del k -renombrado, $k < 2n$.

5.3. Trabajo Futuro

Aún quedan muchas preguntas abiertas. Una de ellas es determinar cual es la característica que debe tener un valor dado n para asegurar que no exista un algoritmo que solucione el k -renombrado, $k < 2n$, sin necesidad de verificar que los coeficientes $\binom{n+1}{1}, \binom{n+1}{2} \dots \binom{n+1}{n}$ no son primos relativos.

Retomemos el Teorema 4.3.10. De manera informal lo que este teorema dice es que para una imagen dividida cromática, orientable, conexa, con estructura binaria simétrica y sin aristas de cruce, el índice siempre tiene la forma $\pm 1 - \sum_{l=0}^{n-1} \binom{n+1}{l+1} k_l$. Sin embargo, la otra dirección no ha sido explorada. Es decir, dados los valores $k'_0, k'_1 \dots k'_{n-1}$, ¿existirá una imagen dividida con las propiedades requeridas y que además su índice de tenga la forma $\pm 1 - \sum_{l=0}^{n-1} \binom{n+1}{l+1} k'_l$?

No obstante, consideramos que la pregunta fundamental es la siguiente. ¿Existe una forma de subdividir \mathbb{K}^5 de forma tal que la subdivisión resultante no contiene 5-simplejos monocromáticos y, además, es una imagen dividida con las mismas propiedades que \mathbb{K}^5 ? Es decir, lo que buscamos es subdividir \mathbb{K}^5 ,

sin tocar la frontera, de forma tal que todos los 5-simplejos monocromáticos desaparezcan. Creemos que es posible diseñar un algoritmo que cumpla con este cometido, ya que de forma muy sencilla se puede desarrollar uno que realiza esta tarea en dimensión 2 y 3. Esto implicaría que cuando se tienen 6 procesadores, el $(2n - 1)$ -renombrado sí tiene solución (ver sección 3.2.4), lo cual sería de gran interés, ya que, además de demostrar que sí existe solución para un problema del cual se pensaba no la tenía, podemos empezar a cuestionarnos si existe una clase de tareas para las cuales su solubilidad está determinada por el número de procesadores que intentan solucionarlas.

Bibliografía

- [1] M. A. Armstrong. *Basic Topology*. Springer-Verlag 1983.
- [2] M. A. Armstrong. *Groups and Symmetry*. Springer-Verlag 1988.
- [3] H. Attiya y S. Rajsbaum. *The Combinatorial Structure of Wait-Free Solvable Tasks*. SIAM Journal on Computing 31(4), pp. 1286-1313 (2002).
- [4] H. Attiya, A. Bar-Noy y D. Dolev. *Sharing Memory Robustly in Message-Passage Systems*. Journal of the ACM 42(1), pp. 124-142 (1995).
- [5] H. Attiya, A. Bar-Noy, D. Dolev, D. Peleg y R. Reischuck. *Renaming in Asynchronous Environment*. Journal of the ACM 37(3), pp. 524-548 (1990).
- [6] H. Attiya y J. Welch. *Distributed Computing: Fundamentals, Simulations and Advanced Topics*. McGraw-Hill 1998.
- [7] E. Borowsky. *Capturing the Power of Resiliency and Set Consensus in Distributed Systems*. Reporte técnico para la Universidad de California (UCLA), 15 de Octubre de 1995.
- [8] E. Borowsky y E. Gafni. *A Simple Algorithmically Reasoned Characterization of Wait-Free Computations*. En: Proceedings of the 16th. Annual ACM Symposium on Principles on Distributed Computing, pp. 189-198 (1997).
- [9] E. Borowsky y E. Gafni. *Generalized FLP Impossibility Result for t -Resilient Asynchronous Computations*. En: Proceedings of the 25th. Annual ACM Symposium on Theory of Computing, pp. 91-100 (1993).
- [10] E. Borowsky y E. Gafni. *Immediate Atomic Snapshots and Fast Renaming*. En: Proceedings of the 12th. Annual ACM Symposium on Principles on Distributed Computing, pp. 41-51 (1993).
- [11] S. Chaudhuri. *More Choices Allow More Faults: Set Consensus Problems in Totally Asynchronous Systems*. Information and Computation 105 (1), pp. 132-158 (1993).
- [12] J. B. Dence y T. P. Dence. *Elements of the Theory of Numbers*. Academic Press 1999.

- [13] E. Gafni, S. Rajsbaum y M. Herlihy. *Subconsensus Tasks: Renaming is Weaker than Set Agreement*. En: Proceedings of the 20th International Symposium on Distributed Computing, pp. 329-338 (2006).
- [14] K. Fan. *Simplicial Maps from an Orientable n -Pseudomanifold into S^m with the Octahedral Triangulation*. Journal of Combinatorial Theory 2, pp. 588-602 (1967).
- [15] M. Fischer, N. Lynch y M. Paterson. *Impossibility of Distributed Consensus with One Faulty Process*. Journal of the ACM 32(2), pp. 374-382 (1985).
- [16] M. Henle. *A Combinatorial Introduction to Topology*. Dover 1994.
- [17] M. Herlihy. *A Tutorial on Algebraic Topology and Distributed Computation*. Notas de curso presentado en la Universidad de California (UCLA), 13 de Diciembre de 1994.
- [18] M. Herlihy y N. Shavit. *The Topological Structure of Asynchronous Computability*. Journal of the ACM 46(6), pp. 858-923 (1999). La primera versión de este artículo aparece en *The Asynchronous Computability Theorem for t -Resilient Tasks*, presentado en: Proceedings of the 12th. Annual ACM Symposium on Principles on Distributed Computing, pp. 111-120 (1993).
- [19] M. Herlihy y S. Rajsbaum. *Algebraic Spans*. Mathematical Structures in Computer Science 10(4), pp. 549-573 (2000).
- [20] N. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers 1996.
- [21] J. R. Munkres. *Elements of Algebraic Topology*. Addison-Wesley 1993.
- [22] M. Saks y Zaharoglou F. *Wait-Free k -Set Agreement is Impossible: the Topology of Public Knowledge*. SIAM Journal on Computing 29(5): pp. 1449-1483 (2000). La primera versión de este artículo fue presentada en: Proceedings of the 25th. Annual ACM Symposium on Theory of Computing, pp. 101-110 (1993).
- [23] *List of important publications in computer science*. En: http://en.wikipedia.org/wiki/List_of_important_publications_in_computer_science (consultado el 29 de octubre de 2007).