Algebraic Spans

(Preliminary Version)

Maurice Herlihy
Computer Science Department
Brown University
Providence RI 02912
herlihy@cs.brown.edu

Sergio Rajsbaum
Digital Equipment Corporation
Cambridge Research Lab
One Kendall Square
Cambridge, MA 02139
rajsbaum@crl.dec.com*

Abstract

Topological methods have yielded a variety of lower bounds and impossibility results for distributed computing. In this paper, we introduce a new tool for proving impossibility results, based on a core theorem of algebraic topology, the *acyclic carrier theorem*, which unifies, generalizes, and extends earlier results.

1 Introduction

Combinatorial and topological methods have yielded a variety of lower bounds results for distributed computing, including general characterizations of the computational power of certain models [4, 12, 13, 14], and the circumstances under which specific problems can be solved [5, 6, 8, 12, 13, 16]. In this paper, we introduce a new tool for proving impossibility results based on a core theorem of algebraic topology. Using the *acyclic carrier theorem* [15, Th. 13.3], we unify, generalize, and extend earlier results. These new proofs are considerably more succinct, so we can present them here in their entirety. Although the mathematical notions underlying this theorem are abstract, they are elementary, being fully covered in the first chapter of Munkres' standard textbook [15].

This paper makes the following contributions.

- Earlier proofs [12, 13] relied on a mixture of combinatorial and continuous arguments. In this paper, we show how to make these proofs completely algebraic, requiring no continuous mathematics. Some important constructs, such as the notion of a *span* [13], are restated in a more elegant algebraic form.
- For each task, set agreement and renaming, we prove a single, short theorem specifying an algebraic property that prevents a protocol from solving the task. These theorems are quite general, yielding results in a variety of models. They imply the known results, and also yield the first impossibility results for renaming using set agreement primitives.

A more complete discussion of related work is postponed to Section 4.

Finally, we believe that these results further illustrate the benefits of formulating concepts and models from distributed computing in the language of algebraic topology, a mature branch of mainstream mathematics.

2 Decision Tasks

Our model is based on [13]. Informally, a *task* is a problem where each process starts with a private input value, communicates with the others by applying operations to shared objects, and halts with a private output value. A *protocol* is a program that solves a task in a concurrent system. A system may be *asynchronous*, placing no constraints on processors' relative speeds, or *synchronous*, requiring processes to run in lockstep. Processes may communicate by applying operations to shared objects, such as read/write memory, or objects with more powerful semantics. They may also communicate by message-passing. A protocol is *t-resilient* if it tolerates failures

* On leave from Instituto de Matemáticas, U.N.A.M., México. Part of this work was done while visiting the Laboratory for Computer Science, MIT. Partly supported by DGAPA Projects.

by t or fewer processes, and it is *wait-free* if it tolerates failures by n out of $n+1$ processes.

Formally, an initial or final state of a process is modeled as a *vertex*, a pair consisting of a process id and a value (either input or output). We think of the vertex as *colored* with the process id. A set of $d+1$ mutually compatible initial or final states is modeled as a *d-dimensional simplex*, (or d-simplex). If the colors of a simplex are all distinct we say that it is *properly colored.* It is convenient to visualize a vertex as a point in Euclidean space, and a simplex as the convex hull of a set of affinely-independent vertexes, the higher-dimensional analogue of a solid triangle or tetrahedron. The complete set of possible initial (or final) states is represented by a set of simplexes, closed under containment, called a *simplicial complex* (or complex). The *dimension* of $\mathcal{C}$ is the dimension of a simplex of largest dimension in $\mathcal{C}$. Where convenient, we use superscripts to indicate dimensions of simplexes and complexes. The k-th *skeleton* of a complex, $skel^k(\mathcal{C}^n)$, is the subcomplex consisting of all simplexes of dimension k or less. The set of process ids associated with simplex S^n is denoted by $ids(S^n)$, and the set of values by $vals(S^n)$.

A *task specification* for $n+1$ processes is given by an input complex $\mathcal{I}^n$, an output complex $\mathcal{O}^n$, and a map Δ carrying each input n-simplex of $\mathcal{I}^n$ to a set of n-simplexes of $\mathcal{O}^n$. This map associates with each initial state of the system (an input n-simplex) the set of legal final states (output n-simplexes).[1] A *solution* to a task is a protocol in which the processes communicate with one another, and eventually halt with mutually compatible decision values.

Figure 1 illustrates the input and output complexes for two-process binary consensus. In general, the input complex for consensus is constructed by assigning independent binary values to $n+1$ processes (this complex is homeomorphic to an n-sphere), and the output complex consists of two disjoint n-simplexes, corresponding to decision values 0 and 1.

Any protocol that solves a task has an associated *protocol complex* $\mathcal{P}$, in which each vertex is labeled with a process id and that process's final state (called its *view*). Each simplex thus corresponds to an equivalence class of executions that "look the same" to the processes at its vertexes.

[1] It is sometimes convenient to extend Δ to simplexes of lower dimension, as in [13]. When $m < n$, $\Delta(S^m)$ is the set of legal final states in executions where only $m+1$ out of $n+1$ processes take steps. This extension does not add any power to the model, since one could capture the same information by adding a flag to each input value indicating whether that process is allowed to participate. This transformation increases the input complex size, but shows that both definitions are equivalent.

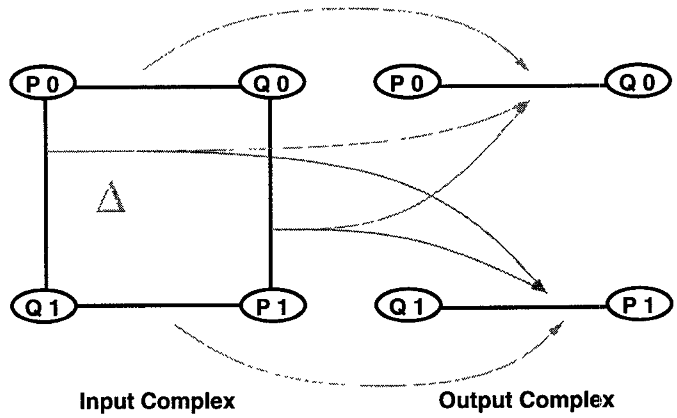

Figure 1: Input and Output Complexes for 2-Process Consensus

For example, consider a model in which synchronous processes communicate by broadcasting messages, but a process can fail in the middle of a broadcast. Figure 2 shows the protocol complex for a three-process single-round protocol in which each process broadcasts its index to the others, and then halts. Each vertex in this figure is a possible final state of a non-faulty process (faulty processes are not shown), and simplexes indicate mutually compatible final states. The labels indicate the messages received: for example, "01?" indicates that messages were received from P and Q, but not R. The central triangle corresponds to the execution in which no process fails: each vertex is labeled with 012. Attached to the central triangle are 1-simplexes corresponding to executions in which one process fails, and disconnected from that triangle are the three 0-simplexes (vertexes) corresponding to executions in which two processes fail.

A *simplicial map* carries vertexes of one complex to vertexes of another so that simplexes are preserved. A simplicial map on properly colored complexes is *color preserving* if it associates vertices of the same color. Let $\mathcal{P}$ be the protocol complex for a protocol. If S^n is an input simplex, let $\mathcal{P}(S^n) \subset \mathcal{P}$ denote the complex of final states reachable from the initial state S^n. A protocol solves a decision task $\langle \mathcal{I}^n, \mathcal{O}^n, \Delta \rangle$ if and only if there exists a color-preserving (i.e., process id-preserving) simplicial map $\delta : \mathcal{P} \to \mathcal{O}^n$, called a *decision map*, such that for every input simplex S^n, $\delta(\mathcal{P}(S^n)) \subset \Delta(S^n)$. We prove our impossibility results by exploiting the topological properties of the protocol complex and the output complex to show that no such map exists.

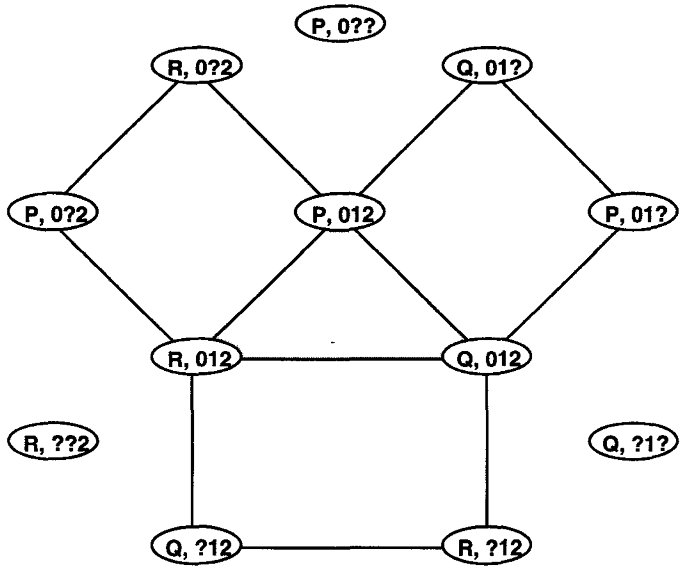

Figure 2: Protocol Complex for One-Round Synchronous Protocol

3 Algebraic Preliminaries

Our discussion closely follows that of Munkres [15, Section 1.13], which the reader is encouraged to consult for more details. The Appendix presents some simple examples of chain complexes and chain homotopies for readers unfamiliar with these notions.

Let $\mathcal{K}$ be an n-dimensional simplicial complex, and $S = (\vec{s}_0, \ldots, \vec{s}_q)$ a q-simplex of $\mathcal{K}$. An *orientation* for S is an equivalence class of orderings on $\vec{s}_0, \ldots, \vec{s}_q$, consisting of one particular ordering and all even permutations of it. For example, an orientation of a 1-simplex $(\vec{s}_0, \vec{s}_1)$ is just a direction, either from $\vec{s}_0$ to $\vec{s}_1$, or vice-versa. An orientation of a 2-simplex $(\vec{s}_0, \vec{s}_1, \vec{s}_2)$ can be either "clockwise," as in $(\vec{s}_0, \vec{s}_1, \vec{s}_2)$, or "counterclockwise," as in $(\vec{s}_0, \vec{s}_2, \vec{s}_1)$. By convention, simplexes are oriented in increasing subscript order unless explicitly stated otherwise.

A *q-chain* of $\mathcal{K}$ is a formal sum of oriented q-simplexes: $\sum_{i=0}^{\ell} \lambda_i \cdot S_i^q$, where λ_i is an integer. When writing chains, we typically omit q-simplexes with zero coefficients, unless they are all zero, when we simply write 0. We write $1 \cdot S^q$ as S^q and $-1 \cdot S^q$ as $-S^q$. We identify $-S^q$ with S^q having the opposite orientation. The q-chains of $\mathcal{K}$ form a free Abelian group $C_q(\mathcal{K})$, called the q-th *chain group* of $\mathcal{K}$. Adjoining the infinite cyclic group Z in dimension -1, $C_{-1}(\mathcal{K}) = Z$.

The *boundary* $\partial_q : C_q(\mathcal{K}) \to C_{q-1}(\mathcal{K})$ is a homomorphism such that

$$\partial_{q-1}\partial_q \alpha = 0,$$

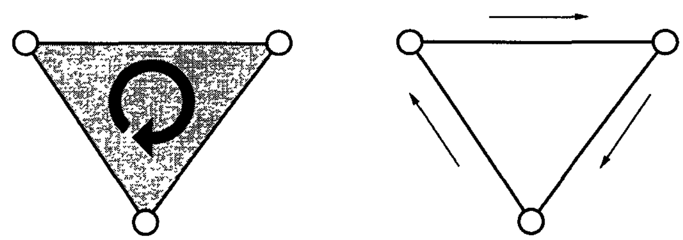

Figure 3: Oriented Simplex and Boundary

and the *augmentation* $\partial_0 : C_0(\mathcal{K}) \to C_{-1}(\mathcal{K})$ is an epimorphism (i.e., a surjective homomorphism).

Let $S^q = (\vec{s}_0, \ldots \vec{s}_q)$ be an oriented q-simplex. Define $face_i(S^q)$, the i^{th} *face* of S^q, to be the $(q-1)$-simplex $(\vec{s}_0, \ldots, \hat{s}_i, \ldots, \vec{s}_q)$, where circumflex denotes omission. The boundary operator $\partial_q : C_q(\mathcal{K}) \to C_{q-1}(\mathcal{K})$, $q > 0$, is defined on simplexes:

$$\partial S^q = \sum_{i=0}^{q} (-1)^i \cdot face_i(S^q),$$

and extends additively to chains: $\partial(\alpha_0 + \alpha_1) = \partial\alpha_0 + \partial\alpha_1$. For $q = 0$, $\partial_0(\vec{s}) = 1$, and extend linearly.[2] (We sometimes omit subscripts from boundary operators.) The boundary operator is illustrated in Figure 3.

A q-chain α is a *boundary* if $\alpha = \partial\beta$ for some $(q+1)$-chain β, i.e., if $\alpha \in Im(\partial_{q+1})$; α is a *cycle* if $\partial\alpha = 0$, i.e., if $\alpha \in ker(\partial_q)$. The group $im(\partial_{q+1})$ is contained in the group $ker(\partial_q)$, and the q^{th} *homology group*[3] is well defined:

$$H_q(\mathcal{K}) = ker(\partial_q)/im(\partial_{q+1}).$$

Informally, if every q-cycle is a boundary, then $\mathcal{K}$ has no "holes" of dimension q, and conversely, any non-boundary q-cycle corresponds to a "hole" of dimension q. If $H_0(\mathcal{K}) = 0$, i.e. is trivial, then $\mathcal{K}$ is connected, and if $H_i(\mathcal{K}) = 0$, then $\mathcal{K}$ has no "holes" of dimension q. If $H_q(\mathcal{K}) = 0$ for every q, we say that $\mathcal{K}$ is *acyclic*.

The *chain complex* $C(\mathcal{K})$ is the sequence of groups and homomorphisms $\{C_q(\mathcal{K}), \partial_q\}$.

Let $C(\mathcal{K}) = \{C_q(\mathcal{K}), \partial_q\}$ and $C(\mathcal{L}) = \{C_q(\mathcal{L}), \partial'_q\}$ be chain complexes for simplicial complexes $\mathcal{K}$ and $\mathcal{L}$. An *augmentation-preserving chain map* (or chain map) ϕ is a family of homomorphisms.

$$\phi_q : C_q(\mathcal{K}) \to C_q(\mathcal{L}),$$

such that $\partial'_q \circ \phi_q = \phi_{q-1} \circ \partial_q$. This identity ensures that chain maps preserve cycles and boundaries. The composition of two chain maps is also a chain map.

[2]Munkres [15] uses ϵ for ∂_0.

[3]Strictly speaking, these are the *reduced* homology groups [15, p.71].

Recall that a simplicial map from $\mathcal{K}$ to $\mathcal{L}$ carries vertexes of $\mathcal{K}$ to vertexes of $\mathcal{L}$ so that every simplex of $\mathcal{K}$ maps to a simplex of $\mathcal{L}$. Loosely speaking, any simplicial map f induces a chain map $f_{\#}$ from $C(\mathcal{K})$ to $C(\mathcal{L})$: when $f(S^q)$ is of dimension q, $f_{\#}(S^q) = f(S^q)$, otherwise $f_{\#}(S^q) = 0$. (We omit subscripts and sharp signs from induced chain maps when the meaning is clear from context.)

If $\phi, \psi : C(\mathcal{K}) \to C(\mathcal{L})$ are chain maps, then a *chain homotopy* from ϕ to ψ is a family of homomorphisms

$$D_q : C_q(\mathcal{K}) \to C_{q+1}(\mathcal{L}),$$

such that

$$\partial'_{q+1} D_q + D_{q-1}\partial_q = \phi_q - \psi_q.$$

Very roughly, if two chain maps are homotopic, then one can be deformed into the other; see Munkres [15] for intuitive justification for this definition.

Remark 3.1 *Let $\phi, \psi : C(\mathcal{K}) \to C(\mathcal{L})$ be chain homotopic maps. Then the chain $(\phi_k - \psi_k - D_{k-1}\partial)(S^k)$ of $C_k(\mathcal{L})$ is a cycle.*

A *symmetry chain map on* $C(\mathcal{K})$, ρ, is the chain map $\rho : C(\mathcal{K}) \to C(\mathcal{K})$ induced by a simplicial map which is a permutation of the vertexes of $\mathcal{K}$. The i-fold composition of ρ is ρ^i. We sometimes abuse notation and also denote the induced simplicial map by ρ. The *orbit* of a simplex S^k consists of all k-simplexes S for which $\rho^i(S^k) = S$, for some i. The k-*orbits* partition the k-simplexes in equivalence classes. For example, if ρ is the identity symmetry, every orbit consists of one simplex.

Let ρ, ρ' be symmetry chain maps on $C(\mathcal{K})$ and $C(\mathcal{L})$, respectively. A chain map $\phi : C(\mathcal{K}) \to C(\mathcal{L})$ is *symmetric with respect to* ρ, ρ', or simply *symmetric*, when ρ, ρ' are understood, if $\rho' \circ \phi = \phi \circ \rho$. Similarly, a chain homotopy D is symmetric if $\rho' \circ D = D \circ \rho$. Notice that any chain map is symmetric with respect to the identity symmetry chain maps.

Definition 3.1 Let ρ, ρ' be symmetry chain maps on $C(\mathcal{K})$ and $C(\mathcal{L})$, respectively. A *symmetric acyclic carrier* from $\mathcal{K}$ to $\mathcal{L}$ is a function Σ that assigns to each simplex S^q of $\mathcal{K}$ a non-empty subcomplex of $\mathcal{L}$ such that (1) $\Sigma(S^q)$ is acyclic, (2) if S^p is a face of S^q, then $\Sigma(S^p) \subset \Sigma(S^q)$, and (3) $\Sigma(\rho(S)) = \rho'(\Sigma(S))$.

A homomorphism $\phi : C_q(\mathcal{K}) \to C_q(\mathcal{L})$ is *carried* by Σ if each simplex appearing with a non-zero coefficient in $\phi(S^m)$ is in the subcomplex $\Sigma(S^m)$.

The next theorem reduces to [15, Th. 13.3], when ρ, ρ' are the identity; the proofs are similar.

Theorem 3.2 (Acyclic Carrier Theorem) *Let Σ be a symmetric acyclic carrier from $\mathcal{K}$ to $\mathcal{L}$.*

(1) *If ϕ and ψ are two symmetric chain maps from $C(\mathcal{K})$ to $C(\mathcal{L})$ that are carried by Σ, then there exists a symmetric chain homotopy of ϕ to ψ that is also carried by Σ.*

(2) *There exists a symmetric chain map from $C(\mathcal{K})$ to $C(\mathcal{L})$ that is carried by Σ.*

Proof: **(1)** By induction. Basis: For each 0-orbit pick a vertex $\vec{s}_0$. Remark 3.1 implies that $(\phi - \psi)(\vec{s}_0)$ is a cycle. Since $\Sigma(\vec{s}_0)$ is acyclic, and ϕ, ψ are carried by Σ, we can choose a 1-chain $D_0(\vec{s}_0)$ carried by Σ, such that $\partial D(\vec{s}_0) = (\phi - \psi)(\vec{s}_0)$. For every $\vec{s}_i \in$ orbit$(\vec{s}_0)$, $\vec{s}_i = \rho^i(\vec{s}_0)$, choose $D_0(\vec{s}_i) = \rho'^i(D(\vec{s}_0))$. Notice that $\partial D_0(\vec{s}_i) = (\phi - \psi)(\vec{s}_i)$, since

$$\begin{aligned} \partial D_0(\vec{s}_i) &= \partial \rho'^i(D(\vec{s}_0)) \\ &= \rho'^i \partial(D(\vec{s}_0)) \\ &= \rho'^i(\phi - \psi)(\vec{s}_0) \\ &= (\phi - \psi)\rho'^i(\vec{s}_0). \end{aligned}$$

Also, $D_0(\vec{s}_i)$ is carried by Σ because Σ is symmetric. Finally, notice that $D_0(\rho(\vec{s}_i)) = \rho'(D_0(\vec{s}_i))$.

For the induction step, assume a symmetric D_j is defined in dimensions j less than d. Pick a representative S_0^d for each d-orbit. Because $\Sigma(S_0^d)$ is acyclic, we can choose a $(d+1)$-chain $D_d(S_0^d)$ such that

$$\partial D(S_0^d) = (\phi - \psi - D\partial)(S_0^d).$$

For each $S_i^d = \rho^i(S_0^d)$ in the same orbit, choose $D(S_i^d) = \rho'^i(D(S_0^d))$. Thus,

$$\begin{aligned} \partial D(S_i^d) &= \partial \rho'^i(D(S_0^d)) \\ &= \rho'^i(\partial D(S_0^d)) \\ &= \rho'^i(\phi - \psi - D\partial)(S_0^d) \\ &= (\phi - \psi - D\partial)(S_i^d) \end{aligned}$$

The last step follows from the induction hypothesis. Finally, it is easy to verify that $D \circ \rho = \rho' \circ D$.

(2) By induction. Basis: For each 0-orbit pick a vertex $\vec{s}_0$. Let $\sigma(\vec{s}_0)$ be a vertex in $\Sigma(\vec{s}_0)$. For $\vec{s}_i \in$ orbit$(\vec{s}_0)$, $\vec{s}_i = \rho^i(\vec{s}_0)$, let $\sigma(\vec{s}_i) = \rho'^i(\sigma(\vec{s}_0))$.

Assume inductively that if $dim(S) < d$, then $\sigma(S)$ is defined and $\partial\sigma(S) = \sigma(\partial S)$. Pick a representative S_0^d for each d-orbit. $\sigma(\partial S_0^d)$ is a well-defined $(d-1)$-chain, and because $\Sigma(S_0^d)$ is acyclic, we can choose a $(d+1)$-chain of $\Sigma(S_0^d)$, $\sigma(S_0^d)$, such that

$$\partial\sigma(S_0^d) = \sigma(\partial S_0^d).$$

For each $S_i^d = \rho^i(S_0^d)$, choose

$$\sigma(S_i^d) = \rho'^i(\sigma(S_0^d)).$$

Hence,

$$\begin{aligned}
\partial\sigma(S_i^d) &= \partial\rho'^i(\sigma(S_0^d)) \\
&= \rho'^i(\partial\sigma(S_0^d)) \\
&= \rho'^i(\sigma(\partial S_0^d)) \\
&= \sigma(\rho'^i(\partial S_0^d)) \\
&= \sigma(\partial S_i^d).
\end{aligned}$$

The penultimate step follows from the induction hypothesis. Finally, $\sigma(S_i^d)$ is carried by Σ because $\sigma(S_0^d)$ is carried by Σ and $\Sigma(\rho(S_0^d)) = \rho'^i(\Sigma(S_0^d))$. ∎

Remark 3.3 *If $\phi, \psi : C(\mathcal{K}) \to C(\mathcal{L})$ are both carried by Σ, and for each S^q in $\mathcal{K}$, $q = dim(S^q) = dim(\Sigma(S^q))$, then $C_{q+1}(\Sigma(S^q)) = 0$, $D_i = 0$ for all i, and ϕ and ψ are equal chain maps.*

4 Algebraic Spans

We can use the acyclic carrier theorem to establish a variety of impossibility results. Our basic strategy is the following. We assume that we have a protocol with complex $\mathcal{P}$ that solves a task $\langle\mathcal{I}, \mathcal{O}, \Delta\rangle$ in a particular model of computation. Let S^ℓ be a simplex, and $\mathcal{S}^\ell$ the complex of its faces. We establish the existence of an acyclic carrier Σ from $\mathcal{S}^\ell$ to $\mathcal{P}$, then the acyclic carrier theorem guarantees the existence of a chain map $\sigma : C(\mathcal{S}^\ell) \to C(\mathcal{P})$, which we call an *algebraic span.* The decision map $\delta : \mathcal{P} \to \mathcal{O}$ is a simplicial map, and therefore induces a chain map $\delta : C(\mathcal{P}) \to C(\mathcal{O})$. The composition of δ and σ is also a chain map:

$$\delta \circ \sigma : C(\mathcal{S}^\ell) \to C(\mathcal{O}).$$

We then use semantic arguments to show that $\mathcal{S}^\ell$ and $\mathcal{O}$ are topologically "incompatible," implying that this chain map cannot exist, and thus deriving a contradiction.

We now discuss how a variety of prior lower bound results can all be given a common reformulation in the language of chain complexes and acyclic carriers. These results are summarized in Figure 4.

Informally, a *subdivision* of a complex is a way of "chopping up" each of its simplexes into smaller simplexes, as illustrated in Figure 5. Any subdivision of a complex has the "same topology" as the original complex, in the sense that the homology groups are unchanged. Much of the earlier work in this area has focused on some notion of subdivision.

Herlihy and Shavit [13] considered wait-free protocols in which $n+1$ processes communicate by reading and writing a shared memory. They showed that it is possible to subdivide the input complex so that there exists a simplicial map, called a *span*, from the subdivision to the protocol complex. (We will refer to this notion of span as a *geometric* span.) They then used the existence of geometric spans to derive a number of impossibility results. For t-resilient protocols, they showed that a geometric span exists on the input subcomplex $\mathcal{I}_t$ containing the vertices colored with process ids $P_0, \ldots, P_t$.

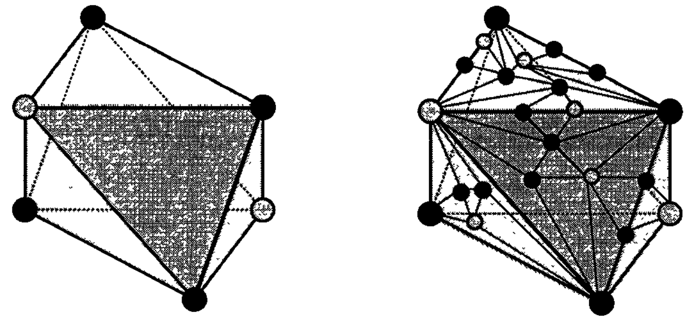

Figure 5: A Subdivided Complex

Herlihy and Rajsbaum [12] considered wait-free protocols using stronger primitives characterized by their ability to solve the (m, j)-agreement task [7], a generalization of consensus [9]. They showed that in this model, a geometric span exists only for the lower-dimensional skeleton of the input complex. In particular, the span is defined on $skel^\ell(\mathcal{I})$, for $\ell \le J(n+1)$, where

$$J(u) = j\left\lfloor\frac{u}{m}\right\rfloor + \min\{j, u \bmod m\} - 1. \qquad (1)$$

The more powerful the primitive, the lower the dimension of the geometric span.

Chaudhuri, Herlihy, Lynch, and Tuttle [8] considered a model in which $n+1$ processes communicate by sending messages over a completely connected network. Computation in this model proceeds in a sequence of rounds. In each round, processes send messages to other processes, then receive messages sent to them in the same round, and then perform some local computation and change state. Communication is reliable, but up to t processes can fail by stopping in the middle of the protocol, perhaps after sending only a subset of their messages. Let $\mathcal{P}_r$ be the protocol complex after r rounds. In their *Bermuda Triangle* construction, they identified a subcomplex of the protocol complex isomorphic to a subdivided simplex. This construction yields a geometric span from an ℓ-simplex S^ℓ to $\mathcal{P}_r$, for $r < \lfloor t/\ell \rfloor$.

Model	Carrier	Span
wait-free read/write objects [13]	Σ_{WF}	$\sigma : C(\mathcal{I}) \to C(\mathcal{P})$
t-resilient read/write objects [13]	Σ_t	$\sigma : C(\mathcal{I}_t) \to C(\mathcal{P})$
wait-free (m, j)-consensus objects [12]	$\Sigma_{m,j}$	$\sigma : C(skel^\ell(\mathcal{I})) \to C(\mathcal{P})$, for $\ell < J(n+1)$
t-resilient synchronous message-passing [8]	Σ_s	$\sigma : C(S^\ell) \to C(\mathcal{P}_r)$, for $r < \ell/t$

Figure 4: Summary of Algebraic Spans

In this paper, we show how these results can be unified by replacing the geometric language of subdivisions and simplicial maps with the more abstract algebraic language of chain complexes and acyclic carriers. To illustrate this remark, we focus first on the wait-free geometric span of Herlihy and Shavit [13]. Establishing the existence of geometric spans required a combination of combinatorial and continuous arguments:

1. $\mathcal{P}(S^m)$ is acyclic,
2. $\mathcal{P}(S^m)$ is simply connected,
3. inductively use these two facts to construct a family of continuous maps from subdivisions of the input complex, and
4. apply simplicial approximation to transform these continuous maps into the desired simplicial maps.

Reformulating this result in algebraic terms yields a simpler derivation: because every $\mathcal{P}(S^m)$ is acyclic, the function Σ_{WF} that assigns to each input simplex S^m the protocol subcomplex $\mathcal{P}(S^m)$ is an acyclic carrier, and the acyclic carrier theorem guarantees the existence of an *algebraic span* $\sigma : C(\mathcal{I}) \to C(\mathcal{P})$, which we use for the impossibility results.

The geometric and algebraic notions of span are related as follows. Any geometric span, reinterpreted as a chain map, is an algebraic span. Although algebraic spans are more abstract, they are simpler in several ways. It is easier to establish the existence of an algebraic span: the second, third, and fourth steps of the derivation are unnecessary. The geometric span is not unique — it is easily seen that there are an infinite number of permissible subdivisions and simplicial maps. By contrast, the acyclic carrier theorem implies that algebraic spans are unique up to chain homotopy.

The other geometric spans have similar reformulations. The t-resilient geometric span corresponds to an acyclic carrier Σ_t from $\mathcal{I}_t$ to $\mathcal{P}$. Each simplex in $\Sigma_t(S^m)$ is labeled with $ids(S^m)$, and corresponds to a t-resilient execution in which processes $ids(S^m) \cup \{P_{t+1}, \ldots, P_n\}$ take steps, but the processes in $\{P_{t+1}, \ldots, P_n\}$ run synchronously. The geometric span for (m, j)-consensus objects corresponds to an acyclic carrier $\Sigma_{m,j}$ from ℓ-simplexes of $\mathcal{I}$ to $\mathcal{P}$, for $\ell \leq J(n+1)$, yielding an algebraic span $\sigma : C(skel^\ell(\mathcal{I})) \to C(\mathcal{P})$. Finally, the Bermuda Triangle construction corresponds to an acyclic carrier Σ_s from an ℓ-simplex S^ℓ to $\mathcal{P}_r$, for $r < \lfloor t/\ell \rfloor$, yielding an algebraic span $\sigma : C(S^\ell) \to C(\mathcal{P}_r)$.

These correspondences are summarized in Figure 4.

Attiya and Rajsbaum [2] take a different approach to proving lower bounds for wait-free read/write memory. Instead of using subdivisions, they use a weaker, purely combinatorial notion called a "divided image," that avoids the need for the geometric and algebraic arguments used to construct geometric spans. Divided images can also be reformulated in terms of chain maps.

5 Set Agreement

In the (N, k)-consensus task [7], each process starts with a private input value from some set *vals*, communicates with the others by applying operations to shared objects, and then halts after choosing a private output value. Each process is required to choose some process's input value, and the set of values chosen should have size at most k. This problem independently was shown to have no t-resilient solution in read/write memory by Borowsky and Gafni [5] and by Herlihy and Shavit [13], and no wait-free solution, by Saks and Zaharoglou [16]. A variety of impossibility results for implementing (N, k)-consensus from (M, j)-consensus were given by Borowsky and Gafni [6], and by Herlihy and Rajsbaum [12].

Theorem 5.1 *Suppose we have a protocol for $(n+1, k)$-set agreement, a properly colored simplex S^ℓ, and an acyclic carrier Σ from S^ℓ to $\mathcal{P}$ such that*

$$vals(\delta(\Sigma(S))) = vals(S) \tag{2}$$

for all simplexes S in S^ℓ. Then $k \geq \ell + 1$.

Proof: Let $\pi : C(\mathcal{O}) \to C(\mathcal{S}^\ell)$ be the chain map induced by the simplicial map sending $\langle P_i, v_j \rangle$ to the vertex of $\mathcal{S}^\ell$ with value v_j. (Here ρ is just the identity symmetry.) The acyclic carrier theorem guarantees that there exists a chain map $\sigma : C(\mathcal{S}^\ell) \to C(\mathcal{P})$ carried by Σ. Let δ be the chain map corresponding to the decision map of $\mathcal{P}$:

$$C(\mathcal{S}^\ell) \xrightarrow{\sigma} C(\mathcal{P}) \xrightarrow{\delta} C(\mathcal{O}) \xrightarrow{\pi} C(\mathcal{S}^\ell)$$

Let $\phi : C(\mathcal{S}^\ell) \to C(\mathcal{S}^\ell)$ be the composition of σ, δ, and π. Let Φ be the acyclic carrier from $\mathcal{S}^\ell$ to itself, $\Phi(\mathcal{S}^i) = \mathcal{S}^i$, and ι the identity chain map on $\mathcal{S}^\ell$. Thus ι is carried by Φ. Equation 2 implies that Φ is an acyclic carrier also for ϕ. Because $dim(\mathcal{S}^i) = i = dim(\Phi(\mathcal{S}^i))$, Remark 3.3 implies that the two maps are equal. Therefore $\iota(\mathcal{S}^\ell) = \phi(\mathcal{S}^\ell) = \mathcal{S}^\ell$.

Assume for contradiction that $k \leq \ell$. In each execution, however, no more than ℓ values are chosen, implying that δ reduces the dimension of every ℓ-simplex. The chain map δ thus sends every ℓ-simplex to the 0 chain, so $\pi \circ \delta \circ \sigma(\mathcal{S}^\ell) = \phi(\mathcal{S}^\ell) = 0$, a contradiction. ∎

Now we consider each of the acyclic carriers described in Section 4.

For asynchronous read/write memory, pick any $S^n \in \mathcal{I}$ with distinct inputs, and consider the acyclic carrier Σ_{WF} described in the previous section. For each $S \in \mathcal{S}^n$, in $\Sigma_{WF}(S^n)$, only the processes in $ids(S)$ take steps, so Equation 2 is satisfied.

Corollary 5.2 *There is no wait-free $(n+1, n)$-consensus protocol in read/write memory [5, 13, 16].*

Consider the acyclic carrier Σ_t for t-resilient read-write memory. One can satisfy Equation 2 by adding a "pre-processing" stage to any $(n+1, t)$-consensus protocol: each process writes it input value to a shared array, waits for $n - t + 1$ values to appear, and replaces its own input with that of the process P_i with smallest index i that writes. No input value from $P_{t+1}, \ldots, P_n$ will ever be chosen.

Corollary 5.3 *There is no t-resilient $(n+1, t)$-consensus protocol in read/write memory [5, 13].*

The acyclic carrier $\Sigma_{m,j}$, when processes share read/write memory and (m, j)-consensus objects, does not directly satisfy Equation 2, because $\Sigma_{m,j}(S)$ may include processes not in $ids(S)$. But in [12] it is shown how to modify the decision values of the processes in $ids(\Sigma_{m,j}(S)) - ids(S)$ so as to satisfy Equation 2 (see [12] for details).

Corollary 5.4 *There is no wait-free $(n+1, J(n+1)-1)$-consensus protocol if processes share a read/write memory and (m, j)-consensus objects.*

The Bermuda Triangle construction [8] satisfies Equation 2 because it associates a unique value with each vertex of S^ℓ, and ensures that the algebraic span for each simplex $(\vec{s}_0, \ldots, \vec{s}_m)$ maps to executions where all input values are taken from the set $\{v_0, \ldots, v_m\}$.

Corollary 5.5 *There is no t-resilient $(n+1, k)$-consensus protocol that takes fewer than $\lfloor t/k \rfloor + 1$ rounds in the synchronous fail-stop message-passing model [8].*

Each of these lower bounds is known to be tight.

6 Renaming

In the renaming task of Attiya *et al.* [1], $n+1$ processes with unique names taken from a large name space must choose unique names taken from a small name space. More precisely, in the $(n+1, K)$-renaming task, the processes are given unique *input names* in the range $0, \ldots, N$, and are required to choose unique *output names* in the range $0, \ldots, K$, where $n \leq K < N$.

To rule out trivial solutions (P_i chooses output name i), we are interested in protocols for which a process's choice is independent of its process id. Let α be a permutation of the process ids. If e is an execution, define $\alpha(e)$ to be the execution in which each occurrence of id P_i is replaced by $\alpha(P_i)$ (i.e., the same interleaving, but processes are renamed). Define $\alpha'\langle P_i, e_i \rangle$ to be $\langle \alpha(P_i), \alpha(e_i) \rangle$. A protocol is *anonymous* if α' is a simplicial map from $\mathcal{P}$ to itself, and $val(\delta\langle P_i, e_i \rangle) = val(\delta(\alpha\langle P_i, e_i \rangle))$ (i.e, both processes choose the same output names in both executions). We restrict out attention to anonymous protocols.

In this section, we use symmetry arguments to give general lower bounds on renaming. We show that if an (n, K)-renaming protocol has a span Σ from a simplex S^ℓ to $\mathcal{P}$ with the property that the protocol behaves "symmetrically" on the boundary of $\Sigma(S^\ell)$, then $K \geq 2\ell + 1$.

Let $S^\ell = (\vec{s}_0, \ldots, \vec{s}_\ell)$ be a simplex where each $\vec{s}_i$ is labeled with process id P_i, and let $\mathcal{S}^\ell$ ($\dot{\mathcal{S}}^{\ell-1}$) be its complex of (proper) faces. Define *rotation maps*

$$\rho : \mathcal{S}^\ell \to \mathcal{S}^\ell$$

by $\rho(\vec{s}_i) = \vec{s}_{i+1 \bmod \ell+1}$, and

$$\rho' : \mathcal{P} \to \mathcal{P}$$

by $\rho'\langle P_i, e_i \rangle = \langle P_{i+1 \bmod \ell+1}, \rho(v_i) \rangle$. Thus the induced chain maps ρ and ρ' are symmetry maps.

The proof is based on the following, purely topological lemma.

Lemma 6.1 *If $\phi : C(\mathcal{S}^n) \to C(\mathcal{S}^n)$ is symmetric, then $\phi(\partial S^n) = k \cdot \partial S^n$, for $k \equiv 1 \pmod{n+1}$.*

Proof: Let $\iota : C(\mathcal{S}^n) \to C(\mathcal{S}^n)$ be the identity chain map. The acyclic carrier theorem implies that there is a symmetric chain homotopy D between ϕ and ι.

In particular, $(\phi - \iota - D\partial)(S_0^{n-1})$ is a cycle of $\mathcal{S}^n$. Since the group of $(n-1)$-cycles of $\mathcal{S}^n$ is infinite cyclic generated by ∂S^n,

$$(\phi - \iota - D\partial)(S_0^{n-1}) = \ell \cdot \partial S^n, \qquad (3)$$

for some integer ℓ.

Note that

$$\rho^i(face_0(S^n)) = (-1)^i face_i(S^n), \qquad (4)$$

and hence

$$\rho(\partial S^n) = \partial S^n. \qquad (5)$$

By definition, $\phi(\partial S^n) = \phi\Sigma_{i=0}^n(-1)^i \cdot face_i(S^n)$. Thus, by Equation 4,

$$\begin{aligned} \phi(\partial S^n) &= \phi\Sigma_{i=0}^n \rho^i face_0(S^n) \\ &= \Sigma_{i=0}^n \phi\rho^i face_0(S^n), \\ &= \Sigma_{i=0}^n \rho^i \phi face_0(S^n) \end{aligned}$$

by symmetry of ϕ. By Equation 3,

$$\phi(\partial S^n) = \Sigma_{i=0}^n \rho^i(\ell \cdot \partial S^n + (\iota + D\partial)(face_0(S^n))).$$

By Equation 5,

$$= \ell(n+1) \cdot \partial S^n + \Sigma_{i=0}^n \rho^i(\iota + D\partial)(face_0(S^n))).$$

By Equation 4 and symmetry of ι,

$$\phi(\partial S^n) = \ell(n+1)\cdot\partial S^n + \Sigma_{i=0}^n(\iota+D\partial)((-1)^i face_i(S^n)).$$

Since ι is the identity,

$$\phi(\partial S^n) = \ell(n+1) \cdot \partial S^n + \partial S^n + D\partial\partial S^n,$$

and the proof follows from $\partial\partial = 0$. ∎

Informally, this lemma says that any map from S^ℓ to itself that is symmetric on the boundary must "wrap" the boundary around itself a non-zero number of times.

Theorem 6.2 *Suppose we have a protocol for $(n+1, K)$-renaming, and a symmetric w.r.t. ρ, ρ' acyclic carrier Σ from $\mathcal{S}^\ell$ to $\mathcal{P}$ such that*

$$ids(\Sigma(S)) = ids(S),$$

for all proper faces S of S^ℓ. Then $K \geq 2\ell + 1$.

Proof: Assume for contradiction that $K < 2\ell + 1$. Let $\pi : \mathcal{O} \to \mathcal{S}^\ell$ be the simplicial map $\pi\langle P_i, v\rangle = \vec{s}_j$, where $j = (P_i + (v \bmod 2)) \bmod \ell + 1$. Let π also denote the induced chain map.

The simplicial map π does not send any ℓ-simplex of $\mathcal{O}(\mathcal{S}^\ell)$ to S^ℓ. This is because π sends an ℓ-simplex of $\mathcal{O}(\mathcal{S}^\ell)$ to S^ℓ only if the processes have chosen all even or all odd output names, which is impossible because the range $0, \dots, 2\ell - 1$ does not contain $\ell + 1$ distinct even or distinct odd names. It follows that the chain map $(\pi)_\ell = 0$.

We have the following sequence of maps.

$$C(\mathcal{S}^\ell) \xrightarrow{\sigma} C(\mathcal{P}) \xrightarrow{\delta} C(\mathcal{O}) \xrightarrow{\pi} C(\mathcal{S}^\ell).$$

Let $\phi : C(\mathcal{S}^\ell) \to C(\mathcal{S}^\ell)$ be the composition of σ, δ, and π. It follows from $(\pi)_\ell = 0$ that $\phi(S^\ell) = 0$, and hence

$$\phi(\partial S^\ell) = 0. \qquad (6)$$

We claim that the chain map ϕ is symmetric. Define the symmetry $\rho'' : \mathcal{O} \to \mathcal{O}$ to be $\rho''\langle P_i, v_i\rangle = \langle P_{i+1 \bmod \ell+1}, v_i\rangle$. We have the following commutative diagram of symmetric chain maps:

$$\begin{array}{ccccccc} C(\mathcal{S}^\ell) & \xrightarrow{\sigma} & C(\mathcal{P}) & \xrightarrow{\delta} & C(\mathcal{O}) & \xrightarrow{\pi} & C(\mathcal{S}^\ell) \\ \rho\downarrow & & \rho'\downarrow & & \rho''\downarrow & & \rho\downarrow \\ C(\mathcal{S}^\ell) & \xrightarrow{\sigma} & C(\mathcal{P}) & \xrightarrow{\delta} & C(\mathcal{O}) & \xrightarrow{\pi} & C(\mathcal{S}^\ell) \end{array}$$

We check that each rectangle commutes: σ is symmetric by the acyclic carrier theorem, δ is symmetric because the protocol is anonymous, and π is symmetric by construction.

Lemma 6.1 implies that $\phi(\partial S^\ell) = k \cdot \partial S^\ell$, for $k \neq 0$, contradicting Equation 6. ∎

Now we consider each of the acyclic carriers described in Section 4. For asynchronous read/write memory, consider the acyclic carrier Σ_{WF} discussed earlier. The carrier for a single input simplex does not satisfy the symmetry requirements of Theorem 6.2. We can, however, construct a symmetric span by "gluing together" the spans from a number of input simplexes as shown in Figure 6. Notice that this complex is a subdivided simplex, and that input names are assigned symmetrically around the boundary.

Definition 6.1 Let $S^n = (\vec{s}_0, \dots, \vec{s}_n)$, where $id(\vec{s}_i) = P_i$. In the *standard chromatic subdivision* of S^n, denoted $\chi(S^n)$, each n-simplex has the form $\{\langle P_0, S_0\rangle, \dots, \langle P_n, S_n\rangle\}$, where S_i is a subsimplex of S^n, such that (1) $P_i \in ids(S_i)$, (2) for all S_i and S_j, one is a subsimplex of the other, and (3) if $P_j \in ids(S_i)$, then $S_j \subseteq S_i$.

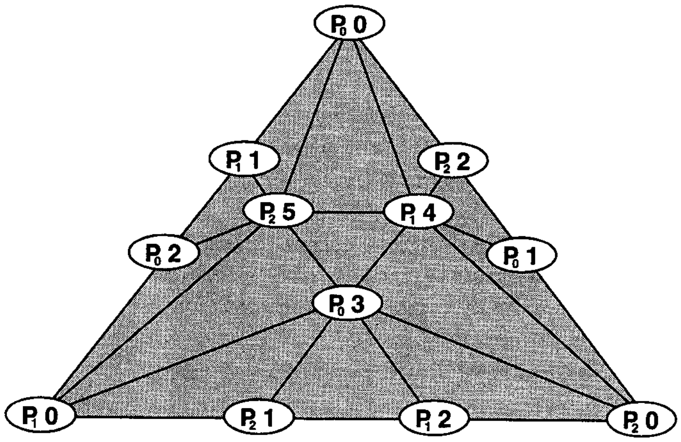

Figure 6: Standard Chromatic Subdivision

We now construct a subdivision $\chi'(S^\ell) \subset \mathcal{I}$, isomorphic to $\chi(S^\ell)$, by assigning input values to vertexes of $\chi(S^\ell)$. The input values are defined inductively. The unique vertex of $\chi'(S^0)$ has input value 0. Assume inductively that we have assigned $m(m+1)/2$ input values to the vertexes in $\chi'(S^{m-1}) = \chi'(\mathit{face}_m(S^m))$. The rotation map that sends P_i to $P_{i+1 \bmod m+1}$ induces a bijective simplicial map $\rho : \mathit{face}_i(S^m) \rightarrow \mathit{face}_{i+1}(S^m)$ by $\rho(\vec{s}_i) = \vec{s}_{i+1 \bmod m+1}$, and also $\rho : \chi(\mathit{face}_i(S^m)) \rightarrow \chi(\mathit{face}_{i+1}(S^m))$, by $\rho\langle P_i, S_i\rangle = \langle\rho(P_i), \rho(S_i)\rangle$. Every vertex $\vec{v} \in \chi(\dot{S}^{m-1})$ is equal to $\rho^i(\vec{u})$, for some $\vec{u} \in \chi(S^{m-1})$. Assign each vertex $\vec{v} \in \chi'(\dot{S}^{m-1})$ the same input value as $\vec{u}$. Finally, the interior vertex $\vec{v}$ of $\chi(S^m)$ labeled with P_i is assigned the input value $m(m+1)/2 + i$.

This construction uses $O(n^2)$ input names. Any renaming protocol for $2n+1$ input names can be transformed into a protocol for a larger number of input names simply by using the shared-memory renaming protocol of Bar-Noy and Dolev [3] to reduce the number of names to $2n+1$, and therefore the impossibility of $(n+1, K)$-renaming for $O(n^2)$ input names implies impossibility for $2n+1$ input names.

Corollary 6.3 *There is no wait-free $(n+1, 2n)$-consensus protocol in read/write memory [11, 13].*

A similar argument yields:

Corollary 6.4 *There is no t-resilient $(n+1, 2t)$-consensus protocol in read/write memory.*

If processes share read/write memory and (m, j)-consensus objects, then it is not known whether $\Sigma_{m,j}$ can be chosen so that $\mathit{ids}(\Sigma_{m,j}(S^\ell)) = \mathit{ids}(S^\ell)$. This condition is clearly satisfied, however, when $m = n+1$, and $j > (n+1)/2$.

Corollary 6.5 *There is no wait-free $(n+1, 2j)$-renaming protocol if processes share a read/write memory and $(n+1, j)$-consensus objects.*

This result is new.

We do not analyze renaming lower bounds for synchronous message-passing systems, since it is known that $\log n$ rounds are necessary and sufficient for wait-free $(n+1, n+1)$-renaming [10] using comparison-based protocols.

Appendix

This appendix gives some simple examples of chain maps and chain homotopies. Readers unfamiliar with chain complexes are encouraged to work out these examples.

A Examples

Let $S^2 = (\vec{s}_0, \vec{s}_1, \vec{s}_2)$ be an oriented 2-simplex (a "solid" triangle), and $\dot{S}^1$ the complex of its proper faces (a "hollow" triangle). $\dot{S}^1$ includes three 0-simplexes (vertexes): $\vec{s}_0$, $\vec{s}_1$, and $\vec{s}_2$, and three 1-simplexes: $S^1_i = \mathit{face}_i(S^2)$, $0 \le i \le 2$ The reader should check that

$$\partial S^1_i = (-1)^i(\vec{s}_{i+1 \bmod 3} - \vec{s}_{i+2 \bmod 3}).$$

The 0-th chain group of $\dot{S}^1$, $C_0(\dot{S}^1)$, is generated by the $\vec{s}_i$, meaning that all 0-chains have the form

$$\lambda_0 \cdot \vec{s}_0 + \lambda_1 \cdot \vec{s}_1 + \lambda_2 \cdot \vec{s}_2,$$

where the λ_i are integers. The first chain group, $C_1(\dot{S}^1)$, is generated by the S^1_i, and all 1-chains have the form

$$\lambda_0 \cdot S^1_0 + \lambda_1 \cdot S^1_1 + \lambda_2 \cdot S^1_2,$$

where the S^1_0 each have standard orientation. Since $\dot{S}^1$ contains no simplexes of higher dimension, the higher chain groups are trivial.

The rotation map $\rho : \dot{S}^1 \rightarrow \dot{S}^1$ defined by $\rho(\vec{s}_i) = \vec{s}_{i+1 \bmod 3}$ induces a chain map $\rho : C(\dot{S}^1) \rightarrow C(\dot{S}^1)$. For $0 \le i < 2$, $\rho(S^1_i) = -S^1_{i+1}$, and $\rho(S^1_2) = S^1_0$. To verify that ρ is a chain map, it suffices to check that

$$\rho(\partial S^1_i) = (-1)^i \rho(\vec{s}_{i+1 \bmod 3} - \vec{s}_{i+2 \bmod 3}) = \partial\rho(S^1_i).$$

The identity map $\iota : \dot{S}^1 \rightarrow \dot{S}^1$ also induces a chain map $\iota : C(\dot{S}^1) \rightarrow C(\dot{S}^1)$. We now show that ι and ρ are chain homotopic, by displaying both an acyclic carrier, and the chain homotopy D. The acyclic carrier Σ is the following: $\Sigma(\vec{s}_i)$ is the complex consisting of S^1_{i-1} and its vertexes, and $\Sigma(S^1_i)$ is the subcomplex of $\dot{S}^1$ containing $\rho(S_i^1)$, $\rho(S_{i+1}^1)$, and their vertexes. Both ι and ρ are carried by Σ, and both $\Sigma(\vec{s}_i)$ and $\Sigma(S_i^1)$ are acyclic (being contractable). The chain homotopy D is given by $D(\vec{s}_i) = (-1)^{i-1}S_{i-1}^1$, and $D(S_i^1) = 0$. It is easily verified that

$$(D\partial + \partial D)(S) = (\iota - \rho)(S).$$

Although every simplicial map induces a chain map, some chain maps are not induced by any simplicial map. Consider the chain map $\phi(\vec{s}_i) = \vec{s}_i, \phi(S_i^1) = S_i^1 + (-1)^i \partial S^2$. Notice that $\phi(\partial S^2) = 4 \cdot \partial S^2$, so this map "wraps" the boundary around itself four times, something no simplicial map could do. This map is not chain homotopic to ι, although $(\phi - \iota)(S)$ is a cycle for every simplex S.

References

[1] H. Attiya, A. Bar-Noy, D. Dolev, D. Peleg, and R. Reischuk. Renaming in an asynchronous environment. *Journal of the ACM*, July 1990.

[2] H. Attiya and S. Rajsbaum. A combinatorial topology framework for wait-free computability. Preprint.

[3] A. Bar-Noy and D. Dolev. Shared memory vs. message passing in an asynchronous distributed environment. In *Proceedings of the 8th Annual ACM Symposium on Principles of Distributed Computing*, pages 371–382, August 1989.

[4] O. Biran, S. Moran, and S. Zaks. A combinatorial characterization of the distributed tasks which are solvable in the presence of one faulty processor. In *Proceedings of the 7th Annual ACM Symposium on Principles of Distributed Computing*, pages 263–275, August 1988.

[5] E. Borowsky and E. Gafni. Generalized flp impossibility result for t-resilient asynchronous computations. In *Proceedings of the 1993 ACM Symposium on Theory of Computing*, May 1993.

[6] E. Borowsky and E. Gafni. The implication of the Borowsky-Gafni simulation on the set consensus hierarchy. Technical Report 930021, UCLA Computer Science Dept., 1993.

[7] S. Chaudhuri. Agreement is harder than consensus: Set consensus problems in totally asynchronous systems. In *Proceedings Of The Ninth Annual ACM Symosium On Principles of Distributed Computing*, pages 311–234, August 1990.

[8] S. Chaudhuri, M.P. Herlihy, N. Lynch, and M.R. Tuttle. A tight lower bound for k-set agreement. In *Proceedings of the 34th IEEE Symposium on Foundations of Computer Science*, October 1993.

[9] M. Fischer, N.A. Lynch, and M.S. Paterson. Impossibility of distributed commit with one faulty process. *Journal of the ACM*, 32(2), April 1985.

[10] M. P. Herlihy and Mark R. Tuttle. Wait-free computation in message-passing systems: Preliminary report. In *Proceedings of the 9th Annual ACM Symposium on Principles of Distributed Computing*, pages 347–362. ACM, August 1990.

[11] M.P. Herlihy. A tutorial on algebraic topology and distributed computation. `http://www.cs.brown.edu/people/mph/home.html`.

[12] M.P. Herlihy and S. Rajsbaum. Set consensus using arbitrary objects. In *Proceedings of the 13th Annual ACM Symposium on Principles of Distributed Computing*, August 1994.

[13] M.P. Herlihy and N. Shavit. The asynchronous computability theorem for t-resilient tasks. In *Proceedings of the 1993 ACM Symposium on Theory of Computing*, May 1993.

[14] M.P. Herlihy and N. Shavit. A simple constructive computability theorem for wait-free computation. In *Proceedings of the 1994 ACM Symposium on Theory of Computing*, May 1994.

[15] J.R. Munkres. *Elements Of Algebraic Topology.* Addison Wesley, Reading MA, 1984. ISBN 0-201-04586-9.

[16] M. Saks and F. Zaharoglou. Wait-free k-set agreement is impossible: The topology of public knowledge. In *Proceedings of the 1993 ACM Symposium on Theory of Computing*, May 1993.