Continuous Tasks and the Asynchronous Computability Theorem

Hugo Rincon Galeana[1]
Embedded Computing Systems Group TU Wien, Austria

Sergio Rajsbaum
UNAM, Instituto de Matemáticas, Mexico City, Mexico

Ulrich Schmid
Embedded Computing Systems Group TU Wien, Austria

Abstract

The celebrated 1999 Asynchronous Computability Theorem (ACT) of Herlihy and Shavit characterized distributed tasks that are wait-free solvable and uncovered deep connections with combinatorial topology. We provide an alternative characterization of those tasks by means of the novel concept of continuous tasks, which have an input/output specification that is a continuous function between the geometric realizations of the input and output complex: We state and prove a precise characterization theorem (CACT) for wait-free solvable tasks in terms of continuous tasks. Its proof utilizes a novel chromatic version of a foundational result in algebraic topology, the simplicial approximation theorem, which is also proved in this paper. Apart from the alternative proof of the ACT implied by our CACT, we also demonstrate that continuous tasks have an expressive power that goes beyond classic task specifications, and hence open up a promising venue for future research: For the well-known approximate agreement task, we show that one can easily encode the desired proportion of the occurrence of specific outputs, namely, exact agreement, in the continuous task specification.

2012 ACM Subject Classification Theory of computation → Distributed computing models; Mathematics of computing → Hypergraphs

Keywords and phrases Wait-free computability, topology, distributed computing, decision tasks, shared memory

Digital Object Identifier 10.4230/LIPIcs.ITCS.2022.73

Related Version *ArXiv Version*: https://arxiv.org/abs/2109.01439

Funding *Hugo Rincon Galeana*: Supported by the Doctoral College on Resilient Embedded Systems at TU Wien and the Austrian Science Fund (FWF) project ByzDEL (P33600).
Sergio Rajsbaum: Supported by UNAM-PAPIIT IN106520. Part of this work was done while visiting École Polytechnique.

1 Introduction

Given a finite set of values V, a *task* $\langle \mathcal{I}, \mathcal{O}, \Delta \rangle$ is a problem where each process of a distributed system starts with a private input value from V, communicates with the others, and halts with a private output value from V. The input complex $\mathcal{I}$ defines the set of possible assignments of input values to the processes, and the output complex $\mathcal{O}$ defines the allowed decisions. The input/output relation Δ specifies, for each input assignment $\sigma \in \mathcal{I}$, a set $\Delta(\sigma) \subseteq \mathcal{O}$ of valid output decisions. Processes are usually *asynchronous*, they can be halted or delayed

[1] Corresponding author.

without warning by cache misses, interrupts, or scheduler pre-emption. In asynchronous systems, it is desirable to design algorithms that are *wait-free*: any process that continues to run will produce an output value, regardless of delays or failures by other processes.

1.1 Background: The Asynchronous Computability Theorem

It is now almost 30 years since Herlihy and Shavit [28] presented the celebrated *Asynchronous Computability Theorem* (ACT). Roughly speaking, the theorem states that a task $\langle \mathcal{I}, \mathcal{O}, \Delta \rangle$ is wait-free solvable in a read/write shared memory distributed system if and only if $\mathcal{I}$ can be subdivided r times, and sent by a simplicial decision map μ to $\mathcal{O}$, respecting the input/output relation (carrier map) Δ. Intuitively, r is the number of rounds the processes need to communicate to solve the task, and hence yields a time complexity characterization as well [29]. The map μ should be *chromatic*, namely, it should preserve process ids.

The ACT reinterprets distributed computing geometrically, and provides an explanation of why some tasks are solvable, and not others. It is particularly useful for proving that some tasks are unsolvable, opening the doors to the very powerful machinery of combinatorial topology; notable examples are the set agreement [28] impossibility result, which generalizes the classic FLP consensus impossibility [12], and the renaming [10] impossibility result. Furthermore, the theorem is the basis for task solvability characterizations in other distributed computing models: systems where at most t processes may crash, synchronous and partially synchronous processes, Byzantine and dependent failures, stronger shared memory communication objects, message passing models and even robot coordination algorithms [2], see Section 5. An overview of topological distributed computing theory, as started by the ACT, can be found in the book [22].

1.2 Motivation: Understanding the characterization provided by the ACT

The ACT provides an explanation of why some tasks are solvable, and not others in terms of the existence of a certain simplicial map. Thus, the intuition behind this explanation has sometimes been presented as the existence of a continuous map, based on the fact that every continuous map between realizations of simplicial complexes is homotopic to the realization of a simplicial map, after subdividing enough. This foundational result in algebraic topology is the *simplicial approximation theorem*, first proved by Brouwer, which served to put homology theory on a rigorous basis [20]. Roughly, the intuitive explanation would say that a task is wait-free solvable if and only if there exists a a continuous map f from $|\mathcal{I}|$ to $|\mathcal{O}|$, respecting Δ, where the decision map μ is a simplicial approximation of f. Unfortunately, the ACT requires μ to be chromatic, and for this reason, this intuitive explanation is wrong.

Indeed, the main source of the technical difficulties in the proof of the ACT is that the objects involved are *chromatic*: each vertex of $\mathcal{I}$ and $\mathcal{O}$ is associated to one of the process ids in the system, and all the simplicial maps are required to preserve vertex ids, i.e., must be *rigid* in that they always preserve the dimension of the simplices. Quite some effort is needed to deal with the resulting requirement of a chromatic decision map μ in the case of a (non-colorless) task such as renaming, which can specify which values can be output by which process. This introduces technical difficulties, which considerably complicated the proofs, cp. [5, 7, 28, 34].

This can be demonstrated by means of the *Hourglass task* shown in Figure 1, where there are three processes denoted by green, red, and yellow. There is no input, hence only one input simplex, and the processes must output labels in $\{0, 1, 2\}$ as shown in the output complex.

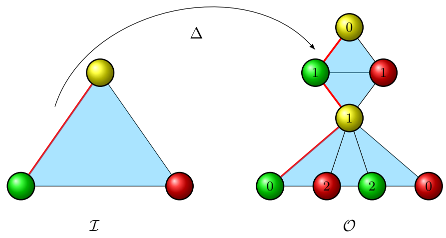


Figure 1 A task that is not colorless: the Hourglass task.

The carrier map Δ defining this task requires (i) to map the corners (solo executions) of $\mathcal{I}$ to the vertices labeled 0 in $\mathcal{O}$, (ii) to map the boundary (executions where two processes participate) of $\mathcal{I}$ to the boundary of $\mathcal{O}$, as shown for the yellow-green boundary in the figure; and (iii) the triangle of $\mathcal{I}$ (executions where all participate), to any triangle in $\mathcal{O}$.

Interestingly, there is a continuous map f from $|\mathcal{I}|$ to $|\mathcal{O}|$ carried by Δ. Yet, it has already been proved in [22] that it is not wait-free solvable. By contrast, as the output complex of the Hourglass task is not link-connected, it is not possible to apply related characterization theorems like [16, Thm. 8.4] to prove this unsolvability (see Section 5 for a more detailed discussion of related work).

1.3 The continuous interpretation of the ACT holds only for colorless tasks

While we are not aware of any formulation of the ACT via continuous functions for general tasks, there is such a continuous version of the ACT for *colorless* tasks (like set agreement), which can be defined without referring to process ids (discussed in more detail in Section 5). It roughly says that a colorless task is wait-free solvable if and only if there is a continuous map f from $|\mathcal{I}|$ to $|\mathcal{O}|$ carried by Δ, where $|\mathcal{K}|$ denotes the geometric realization of the complex $\mathcal{K}$. Since all involved complexes are colorless here, the proofs are much simpler. Indeed, the colorless version of the ACT [22, Ch. 4] is essentially a distributed computing version of the simplicial approximation theorem. In the case of the Hourglass task, for example, the simplicial approximation theorem says that there is some r such that there is a simplicial map μ from $\mathrm{Bary}^r(\mathcal{I})$ to $\mathcal{O}$ respecting Δ. However, this does not result in a wait-free algorithm, since there is no such μ that preserves colors, i.e., is rigid.

1.4 In search of a chromatic version of the Simplicial Approximation Theorem

In order to fully understand the meaning of the geometric reinterpretation of distributed computing provided by the ACT for general tasks, a chromatic version of the simplicial approximation theorem would be needed.

Whereas several constructions of a chromatic simplicial approximation have been used in the existing proofs of the ACT [5, 7, 28, 34] and some generalizations [16, 36], they are complicated and tailored to a specific context, namely, link-connected output complexes resp. the images of the carrier map Δ. We are not aware of a simplicial approximation theorem that could guarantee a chromatic map μ under more general general conditions, like the ones for our continuous tasks (see Definition 9).

1.5 Contributions

Theorem 1 below re-states one direction of the ACT (for reference purposes); it actually holds in the other direction as well. It is the easier if-direction, and can be proved in several ways, see e.g. [5, 7, 28, 34], essentially by considering an appropriate subset of all wait-free executions of a full-information protocol: in any wait-free read/write model, any protocol solving a given task induces a subdivision of the task's input complex.

▶ **Theorem 1** (ACT (if-direction) [28]). *If* $\langle \mathcal{I}, \mathcal{O}, \Delta \rangle$ *is solvable, then there exists a chromatic subdivision* $\mathrm{Sub}(\mathcal{I})$ *and a chromatic simplicial map* $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ *carried by* Δ.

Our paper is concerned with the only-if direction of Theorem 1. It has three main contributions: (1) we introduce the novel notion of a continuous task and show that it allows for more expressive task specifications, (2) we formulate and prove a chromatic version of the simplicial approximation theorem, and (3) we use these notions to formulate a continuous version the ACT, denoted CACT, which we prove to be equivalent to the ACT in the wait-free shared memory model. Rather than focusing on a specific model of computation, however, the only property we require from the model is that Theorem 1 holds, i.e., if some protocol solves a task, then the protocol determines a chromatic subdivision of the input complex. We refer to any such model by ASM, and give examples of such models (such as the read-write shared memory model of the original ACT [28] and the *Iterated Immediate Snapshot* (IIS) model) in Section 5.

In more detail, our paper contains the following contributions:

(1) We introduce a *continuous task* as a triple $\langle \mathcal{I}, \mathcal{O}, f \rangle$: The possible input and output configurations are determined by the chromatic input and output complexes $\mathcal{I}, \mathcal{O}$, as in the usual task notion. The input/output specification f is a continuous function from $|\mathcal{I}|$ to $|\mathcal{O}|$, instead of an input/output relation Δ. We identify an additional property called *chromatic* for f, which intuitively requires f to satisfy a minimal color and local dimension preserving requirement, stated formally in Definition 5.
Semantically, continuous tasks provide interesting additional expressiveness facilities with respect to traditional input/output specifications in the form of a carrier map Δ. Indeed, they open up various interesting research questions, like limits of expressiveness and decidability of continuous tasks, which we cannot even touch here. In Section 4, however, we introduce refined versions of the well-known 1/3-approximate agreement task, which demonstrate that constraints on the *density*, i.e., "likelihood" of occurrence, of specific outputs for a given input can be expressed by means of continuous task specifications.

(2) We state and prove Theorem 22, a chromatic version of the simplicial approximation theorem, for chromatic functions (see Definition 5), which may be of independent interest also. In a way, it off-loads part of the complexity of constructing geometric subdivisions that ensure rigid maps for an *arbitrary* continuous function f to the definition of a chromatic function f.

(3) Using our chromatic simplicial approximation Theorem 22, we prove that chromatic functions precisely capture the notion of solvability in an ASM model. This leads to our CACT Theorem 24, which states that a task $\langle \mathcal{I}, \mathcal{O}, \Delta \rangle$ is wait-free solvable if and only if there exists a corresponding continuous task $\langle \mathcal{I}, \mathcal{O}, f \rangle$. Finally, Theorem 25 states that any continuous task is solvable, which implies the converse of Theorem 1.

Overall, our results provide a refined explanation on the reasons of why a task may or may not be solvable in an ASM model, and provide an alternative perspective and proof of the ACT (Section 5 discusses several related theorems).

Paper organization. In Section 2, we define continuous tasks and the meaning of solving such a task, and present the chromatic simplicial approximation theorem. Section 3 contains our CACT theorem, and Section 4 is devoted to the application showing the expressive power of continuous tasks. A discussion of additional related work and some conclusions and directions of future research are provided in Section 5 and Section 6, respectively. In the appendix, we provide a collection of background combinatorial topology and distributed computing definitions for ease of reference, primarily taken from [22].

2 Continuous Tasks and the Chromatic Simplicial Approximation Theorem

In this section, we extend the standard language (see e.g. [22]) of combinatorial topology in distributed computing to be able to specify continuous tasks. More specifically, given arbitrary geometric realizations $|\mathcal{I}|$ and $|\mathcal{O}|$, i.e., metric topological spaces formed as the union of geometric simplices $|\sigma|$ corresponding to the respective abstract simplices in $\mathcal{I}$ and $\mathcal{O}$, we introduce the concept of a chromatic function, which is a continuous function that plays the role of a minimal carrier map that determines task solvability.

2.1 Chromatic functions and continuous task solvability

We first need to extend the notion of a coloring in order to assign a set of colors to a point in the geometric realization $|\mathcal{K}|$ of a finite simplicial complex $\mathcal{K}$. We start with the carrier $\mathrm{carr}(x, \mathcal{K})$ of a point x in the geometric realization $|\mathcal{K}|$ of a simplicial complex $\mathcal{K}$.

▶ **Definition 2.** *For a point $x \in |\mathcal{K}|$, let the carrier of x, denoted $\sigma = carr(x, \mathcal{K})$, be the unique smallest simplex $\sigma \in \mathcal{K}$ such that $x \in |\sigma|$. We can also define the carrier of a set $S \subseteq |\mathcal{K}|$, as $carr(S, \mathcal{K}) = \bigcup_{x \in S} carr(x, \mathcal{K})$.*

In Figure 2, the carrier of interior point x is the whole triangle, while the carrier of x' is just the green-red edge.

▶ **Definition 3** (Extended coloring). *Let $\mathcal{K}$ be an m-dimensional simplicial complex with a coloring $\chi : \mathcal{K} \to 2^{\Pi} = 2^{\{1,\dots,m+1\}}$. We define the extended coloring $|\chi| : |\mathcal{K}| \to 2^{\Pi}$ with respect to $\mathcal{K}$ as $|\chi|(x) = \chi(carr(x, \mathcal{K}))$ for every $x \in |\mathcal{K}|$.*

In Figure 2, the extended coloring of x is the set of all colors, while the extended coloring of x' consists of the red and green colors.

In addition to defining an extended coloring, we also need a notion of closeness. In topology, a neighborhood of a point x is a collection of open sets that include x and defines what does it mean to be close to x. We will use the following Definition 4, illustrated in Figure 2.

▶ **Definition 4** (Simplicial neighborhood). *Let $\mathcal{I}$ be an abstract simplicial complex, $x \in |\mathcal{I}|$ a point in its geometric realization, and $\sigma = \mathrm{carr}(x, \mathcal{I})$. We say that $N \subseteq |\sigma|$ is a simplicial neighborhood of x if $N \cong |\{x_1, x_2, \dots, x_r\}|$, where each $x_i \in |\sigma|$, and $x \in \mathrm{int}(N)$ is a point in its interior. That is, N is homeomorphic ($\cong$) to a geometric simplex generated from points that belong to $|\sigma|$.*

A chromatic function f prohibits mapping a neighborhood to a neighborhood of smaller dimension, and guarantees that the color map is preserved.

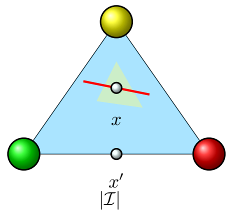


Figure 2 The inner (yellow) triangle, and the red segment are neighborhoods of $x \in \text{int}(|\sigma|)$ of dimension 2 and 1, respectively, in the outer (blue) triangle $|\sigma|$.

▶ **Definition 5** (Chromatic Function)**.** *We say that a continuous function $f : |\mathcal{I}| \to |\mathcal{O}|$ is chromatic (with respect to $\mathcal{I}$ and $\mathcal{O}$) if, for every $x \in |\mathcal{I}|$, and any simplicial neighborhood N of x, it guarantees $|\chi|(N) \cap |\chi|(f(N)) \geq \dim(N) + 1$.*

We note that not every continuous function is chromatic. An example is shown in Figure 3, where we assume that f maps the entire red curve in the blue simplex $|\sigma| \in |\mathcal{I}|$ on the left to the central red vertex in $|\mathcal{O}|$ on the right. This collapses a 1-dimensional neighborhood N of x to a 0-dimensional neighborhood $f(N)$. Similarly, if we assume that f also maps the whole area above this curve to the line connecting the central red vertex to the small yellow vertex on the boundary of the upper simplex in $|\mathcal{O}|$, it also collapses a 2-dimensional neighborhood to a 1-dimensional neighborhood. Finally, f also violates the color preservation requirement, as a 1-dimensional neighborhood $N' \in |\rho|$ of x' lying in the face $|\rho| \in |\sigma|$ consisting of the (green, yellow) edge is mapped to $f(N')$, which lies on the the (red, green) edge in the upper simplex in $|\mathcal{O}|$.

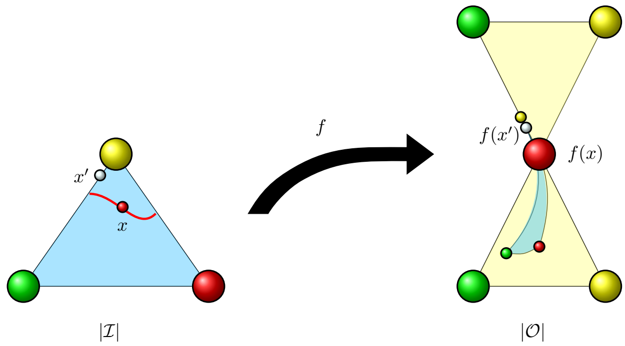


Figure 3 f is non-chromatic, since it maps a 1-dimensional neighborhood of a point x (the red curve) with $\text{carr}(x, \mathcal{I}) = \{\text{red}, \text{green}, \text{yellow}\}$ to the 0-dimensional red central vertex. It also violates the color preservation requirement, as a 1-dimensional neighborhood of x' within the (green, yellow) edge in $|\mathcal{I}|$ is mapped onto the (red, green) edge in the upper simplex in $|\mathcal{O}|$.

Lemma 6 and Corollary 7 show that it is possible to chromatically subdivide the input complex while maintaining the color preservation property of a given chromatic function f. Indeed, even if some chromatic subdivision does not fit w.r.t. color preservation, we can make an arbitrarily small perturbation to the vertices in the subdivision fitting. We note that the proof of this lemma is substantially less involved than the perturbation argument in the proof of the ACT [28], [22, Ch. 11]. In fact, rather than constructing subdivisions that ensure rigidity for arbitrary continuous functions f, we only need subdivisions that allow f to remain a chromatic function.

▶ **Lemma 6.** *Let* $f : |\mathcal{I}| \to |\mathcal{O}|$ *be a chromatic function with respect to* $\mathcal{I}$ *and* $\mathcal{O}$*. For any 1-layer chromatic subdivision* $\mathrm{Ch}(\mathcal{I})$ *of* I*, there exist a geometric realization* $|\mathrm{Ch}(\mathcal{I})|$ *such that* $f : |\mathrm{Ch}(\mathcal{I})| \to |\mathcal{O}|$ *is chromatic with respect to* $\mathrm{Ch}(\mathcal{I})$ *and* $\mathcal{O}$*.*

Proof. To prove the lemma, it suffices to show that f is chromatic for any given face $\mathrm{Ch}(\sigma)$ of dimension k. For $k = 0$, this is trivial, so assume that we have shown this already for all faces of dimension $k-1$, and consider a face $\mathrm{Ch}(\sigma)$ with $\dim(\mathcal{I}) = k \geq 1$. Let $x \in \mathrm{int}(|\sigma|)$ be an internal point of $|\sigma|$. Since f is chromatic with respect to $\mathcal{I}$, there exists a k-dimensional simplicial neighborhood $N \subseteq \mathrm{int}(|\sigma|)$ of x such that $|\chi|(f(N)) = \{1, \dots, k+1\}$, i.e., includes all colors. We will first show that there is a point $y \in f(N)$ such that $|\chi|(y) = \{1, \dots, k+1\}$: Assuming the contrary, $f(N)$ would be contained in the $k-1$ skeleton of $|\mathcal{O}|$. Consequently, also $f(x)$ does not include all colors, i.e., there exists a color $i \notin |\chi|(f(x))$. Since the $k-1$ skeleton of $|\mathcal{O}|$ that includes color i is a closed set that does not include $f(x)$, there is a k-dimensional ε-ball $B_\varepsilon(f(x))$ around $f(x)$ that does not intersect the $k-1$ skeleton of $|\mathcal{O}|$ that includes i as a color. Due to continuity of f, there exists a k-dimensional δ-ball $B_\delta(x)$ around x that is contained in N, which is mapped to $B_\varepsilon(f(x))$. Since every point in $N \subseteq \mathrm{int}(|\sigma|)$ has all $k+1$ colors in its carrier, we also obtain $|\chi|(B_\delta(x)) = k+1$, but $|\chi|(f(B_\delta(x))) \leq k$, since it does not include i. This contradicts that f is chromatic.

Therefore, there exists a point $y \in f(N)$ such that $|\chi|(y) = \{1, \dots, k+1\}$. Since y is hence an interior point in $|\mathcal{O}|$, there exists $B_\varepsilon(y) \subseteq |\mathcal{O}|$ where every point has all colors in its carrier. Since $y \in f(N)$ and $N \subseteq \mathrm{int}(|\sigma|)$, there exists $z \in \mathrm{int}(|\sigma|)$ such that $f(z) = y$. Again from continuity of f, it follows that there exists a δ-ball $B_\delta(z)$ around z that maps to $f(B_\delta(z)) \subseteq B_\epsilon(y)$, where every point in the latter has all colors in its carrier. Therefore, in the to-be-constructed subdivision $\mathrm{Ch}(\sigma)$, we can create the new central face σ' of dimension k inside $B_\delta(z)$, which will make f chromatic with respect to $|\sigma'|$.

According to the induction hypothesis, we have already subdivided the boundary of σ (with dimension $< k$) in a way that makes f chromatic. What still remains to be done, however, is to show that f is also chromatic in the non-central faces of dimension k generated by our subdivision. For this purpose, it suffices to show that every $k-1$-dimensional face ρ' that includes at least one vertex from σ' can be chosen such that it includes all its k colors $\{1, \dots, k\}$. This suffices, since lower-dimensional faces of ρ' inherently preserve the chromatic property of f: (1) For lower-dimensional faces on the boundary of σ, this follows from the induction hypothesis. (2) For $k-2$-dimensional faces in the interior of σ that originate from the intersection $\rho' \cap \rho''$ with another $k-1$-dimensional face ρ'', the intersection of their color sets must contain $k-2$ colors, which guarantees that f is chromatic also on $\rho' \cap \rho''$. Finally, (3) for lower-dimensional faces lying in σ', the color set even comprises all k colors and cannot hence make f non-chromatic.

To finally justify that, for any $k-1$-dimensional face $\rho' = (v_1, \dots, v_k)$ of vertices in our subdivision, there must indeed be a simplicial neighborhood N'_{k-1} that connects these vertices in N such that $|\chi|(N'_{k-1}) = \{1, \dots, k\}$, we just repeat the argument in our first step above: Assuming the contrary, $f(N'_{k-1})$ would be in the $k-2$-skeleton of $|\mathcal{O}|$, which does not include $\{1, \dots, k\}$ and would hence lead to a contradiction to f being chromatic with respect to $\mathcal{I}$ and $\mathcal{O}$. Consequently, we can always find a suitable choice for the geometric realization for ρ', namely N'_{k-1}. According to case (2) above, for neighboring faces ρ' and ρ'', we can choose the common border $\rho' \cap \rho''$ arbitrarily within $N'_{k-1} \cap N''_{k-1}$.

This completes the proof, since we showed that we can insert interior faces and connect new vertices in the geometric subdivision without making f non-chromatic. ◀

▶ **Corollary 7.** *Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a chromatic function with respect to $\mathcal{I}$ and $\mathcal{O}$. For any chromatic subdivision* $\mathrm{Sub}(\mathcal{I})$ *of $\mathcal{I}$, there exists a geometric realization* $|\mathrm{Sub}(\mathcal{I})|$ *of such that* $f : |\mathrm{Sub}(\mathcal{I})| \to |\mathcal{O}|$ *is chromatic with respect to* $\mathrm{Sub}(\mathcal{I})$ *and $\mathcal{O}$.*

Our first step towards establishing a discrete/continuous duality between chromatic functions and chromatic simplicial maps, i.e., simplicial maps that preserve the colors of vertices (and hence are rigid), is showing that the geometric realization of a chromatic simplicial map is a chromatic function.

▶ **Lemma 8.** *Let $\mu : \mathrm{Ch}^k(\mathcal{I}) \to \mathcal{O}$ be a chromatic simplicial map. Then, $|\mu| : |\mathcal{I}| \to |\mathcal{O}|$ is a chromatic function.*

Proof. Continuity follows from $|\mu|$ being an affine mapping from $|\mathrm{Ch}^k(\mathcal{I})| \cong |\mathcal{I}| \to |\mathcal{O}|$.

To show that it is a chromatic function, we start with the color preservation property. Notice that, since μ is simplicial and chromatic, $\chi(\sigma) = \chi(\mu(\sigma))$ for any simplex $\sigma \in \mathcal{I}$. This implies that, for any $x \in |\mathcal{I}|, |\chi|(x) = |\chi|(|\mu|(x))$. Moreover, it follows that, for any point x and any simplicial neighborhood N of x of dimension r, $|\chi|(N) = |\chi|(|\mu|(N))$. Since σ must be of dimension at least r since N has dimension r, it follows that $\big||\chi|(N)\big| = \big||\chi|(N) \cap |\chi|(|\mu|(N))\big| \geq r + 1$. ◀

We further develop the discrete/continuous duality by defining continuous tasks. Since chromatic functions correspond to chromatic simplicial maps, and chromatic simplicial maps determine task solvability, we can also use chromatic functions to express solvable tasks.

▶ **Definition 9** (Continuous task). *We say that a triple $\langle \mathcal{I}, \mathcal{O}, f\rangle$ is a continuous task if $\mathcal{I}$ and $\mathcal{O}$ are pure chromatic simplicial complexes of the same dimension, and $f : |\mathcal{I}| \to |\mathcal{O}|$ is a chromatic function.*

In order for our task definition to be complete, we also need to define a criterion for task solvability. A continuous task is solvable in ASM if there exists an algorithm $\mathcal{A}$ with a decision map μ that "approximates" the chromatic function of the continuous task. Therefore we first need a formal definition for a chromatic approximation.

▶ **Definition 10** (Chromatic approximation). *Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a chromatic function. We say that a chromatic simplicial map $\mu : Sub(\mathcal{I}) \to \mathcal{O}$ is a chromatic approximation to f if, for all $\sigma \in Sub(\mathcal{I})$, $\mu(\sigma) \subseteq \mathrm{carr}(f(|\sigma|), \mathcal{O})$.*

Notice that while a chromatic approximation of f is a more relaxed definition than a simplicial approximation of f in the topological sense, as $f(\mathrm{int}(|\sigma|)) \subseteq \mathrm{star}^o(\mu(\sigma))$ need not hold, it adds a color preservation constraint.

▶ **Definition 11** (ASM Continuous Task Solvability). *We say that an algorithm $\mathcal{A}$ in ASM solves a continuous task $T = \langle \mathcal{I}, \mathcal{O}, f\rangle$, if $\mathcal{A}$ induces a subdivision $Sub(\mathcal{I})$ of $\mathcal{I}$, and a (simplicial) decision map $\mu : Sub(\mathcal{I}) \to \mathcal{O}$ that is a chromatic approximation, i.e., for each $\sigma \in Sub(\mathcal{I}), \mu(\sigma) \subseteq \mathrm{carr}(f(|\sigma|), \mathcal{O})$.*

Another way to formulate this condition is by defining an induced task Δ_f associated to f.

▶ **Definition 12** (Induced Task). *Given a continuous task $T = \langle \mathcal{I}, \mathcal{O}, f\rangle$, we define the task induced by T as $T_f = \langle \mathcal{I}, \mathcal{O}, \Delta_f\rangle$, where $\Delta_f : \mathcal{I} \to 2^{\mathcal{O}}$ is the carrier map induced by f as given by $\Delta_f(\sigma) = \{carr(f(x), \mathcal{O}) \mid x \in |\sigma|\} = carr(f(|\sigma|), \mathcal{O})$.*

Recall from the ACT Theorem 1 that T_f is solvable in ASM if there exists a subdivision $\mathrm{Sub}(\mathcal{I})$ and a decision map $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ carried by Δ_f. According Definition 11, this indeed implies continuous task solvability for T as well.

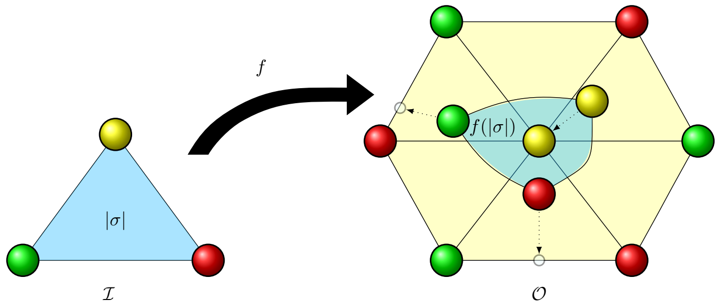


Figure 4 The interior of a simplex is mapped to an open star of a vertex. All yellow vertices are mapped to the center of the star.

2.2 The chromatic simplicial approximation theorem

In the previous subsection, we provided the motivation and definitions for chromatic functions and continuous tasks as part of a discrete/continuous duality for ASM. We showed that chromatic simplicial maps generate chromatic functions. However, in order for this correspondence to be complete, we need to show that we can approximate any chromatic function with a simplicial chromatic map. In general algebraic topology, the simplicial approximation theorem allows us to discretize continuous functions. In the context of distributed computing, however, we cannot apply the simplicial approximation theorem directly, since it does not necessarily preserve the color structure.

Therefore, in this subsection, we will prove that any chromatic function from a geometric pure simplicial complex $|\mathcal{I}|$ into a geometric pure simplicial complex $|\mathcal{O}|$ of the same dimension admits a chromatic approximation. To this end, we introduce the notion of the chromatic projection for some color c. It maps interior points of a geometric simplex $|\sigma|$ to its border, by taking the ray r from the vertex $v_i \in V(\sigma)$ with color c to x, and mapping x to the intersection of r and the opposite border of $|\sigma|$.

▶ **Definition 13** (Chromatic projection)**.** *Let $\mathcal{I}$ be a pure chromatic simplicial complex, and $c \in \chi(\mathcal{I})$. We define the chromatic projection with respect to c as $\pi_c : |\mathcal{I}| \setminus |\chi|^{-1}(\{c\}) \to |\mathcal{I} \setminus \chi^{-1}(\{c\})|$ as follows: For $x \in |\mathcal{I}| \setminus |\chi|^{-1}(\{c\})$, let $\sigma = \mathrm{carr}(x, \mathcal{I}) = \{v_1, \ldots, v_n\}$ be the carrier of x in $|\mathcal{I}|$. If $c \notin \chi(\sigma)$, we define $\pi_c(x) = x$. Now, assume that $\chi(v_i) = c$, in which case we must have $n \geq 2$. Writing $x = \sum_{k=1}^{n} \alpha_k \cdot v_k$, where each $v_k \in \sigma$, and the α_k's correspond to the affine coordinates of x with respect to σ, we define $\pi_c(x) = \sum_{k \neq i} \frac{\alpha_k}{1 - \alpha_i} \cdot v_k$.*

An example for a chromatic projection can be found in Figure 4, for $c = \text{yellow}$: Both the points x marked by the green and the red inner node on the boundary of $f(|\sigma|)$ are mapped to the respective border of their carriers that lies opposite of the yellow vertex.

Figure 4 also illustrates the pivotal concept of star-covering introduced in Definition 14, which requires the image of the interior of a simplex $|\sigma|$ to be contained in the open star $\mathrm{star}^o(w)$ of some vertex $w \in V(\mathcal{O})$.

▶ **Definition 14** (Star-covered subdivision)**.** *Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a chromatic function. We say that a chromatic subdivision Sub($\mathcal{I}$) is star-covered with respect to f if for any $\sigma \in \mathrm{Sub}(\mathcal{I})$, $f(\mathrm{int}(|\sigma|)) \subseteq \mathrm{star}^o(w)$ for some $w \in V(\mathcal{O})$.*

The following Lemma 15 shows that a sufficiently deep chromatic subdivision guarantees that a chromatic function f will be star-covered with respect to f.

▶ **Lemma 15.** *Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a chromatic function. Then, there exists $k \in \mathbb{N}$ such that $\mathrm{Ch}^k(\mathcal{I})$ is star-covered with respect to f.*

Proof. Since $\mathcal{O}$ is pure and of the same dimension as $\mathcal{I}$, the collection $\mathcal{C} = \{\mathrm{star}^o(w) \mid w \in \mathcal{O}\}$ is an open covering of $|\mathcal{O}|$. We claim that $\mathcal{C}^{-1} = \{f^{-1}(U \cap f(|\mathcal{I}|)) \mid U \in C\}$, the collection of preimages of C under f is a finite open cover for $|\mathcal{I}|$. Indeed, for every $x \in |\mathcal{I}|$, $f(x) \in |\mathcal{O}|$. Since $\mathcal{C}$ is an open cover for $|\mathcal{O}|$, there exists $U \in \mathcal{C}$ such that $f(x) \in U$. Since U is open and f is continuous, the preimage of a sufficiently small neighborhood $N(f(x)) \subseteq (U \cap f(|\mathcal{I}|))$ is open and hence contained in $\mathcal{C}^{-1}$.

Since $\mathcal{I}$ is a finite simplicial complex, $|\mathcal{I}|$ is a compact metric space. Therefore, there exists a Lebesgue number $\epsilon > 0$ such that any set $S \subseteq |\mathcal{I}|$ with a diameter less than ϵ is contained in an element of $\mathcal{C}^{-1}$. Since the chromatic subdivision is a mesh shrinking operation on $|\mathcal{I}|$, there exists some $k \in \mathbb{N}$ such that any $|\sigma|$ in $|\mathrm{Ch}^k(\mathcal{I})|$ has a diameter less than ϵ. Therefore, for any $\sigma \in \mathrm{Ch}^k(\mathcal{I})$, $|\sigma| \in S$ for some $S \in \mathcal{C}^{-1}$. Consequently, $f(|\sigma|) \subseteq \mathrm{star}^o(w)$ for some $w \in V(\mathcal{O})$, which confirms that $\mathrm{Ch}^k(\mathcal{I})$ is indeed star-covered with respect to f. ◀

A star-covered subdivision $\mathrm{Sub}(\mathcal{I})$ induces a coloring for the facets $\sigma \in \mathrm{Sub}(\mathcal{I})$. Definition 16 simply associates a facet σ with the colors of all star centers that cover $f(\sigma)$ in $\mathcal{O}$. For example, in Figure 4, σ is assigned the yellow color here. This color assignment will be fundamental to the proof of Theorem 22.

▶ **Definition 16** (Star coloring). *Let $\mathrm{Sub}(\mathcal{I})$ be a star-covered subdivision with respect to $f : |\mathcal{I}| \to |\mathcal{O}|$. For $\sigma \in \mathcal{I}$, we define $\chi^*(\sigma) = \{c \mid w \in V(\mathcal{O}),\ \chi(w) = c,\ f(\mathrm{int}(|\sigma|)) \subseteq \mathrm{star}^o(w)\}$.*

▶ **Definition 17.** *Let $\mathrm{Sub}(\mathcal{I})$ be a star-covered subdivision with respect to $f : |\mathcal{I}| \to |\mathcal{O}|$ and $c \in \chi(\mathcal{O})$. We say that a facet $\sigma \in \mathrm{Sub}(\mathcal{I})$ is c-covered if $c \in \chi^*(\sigma)$, and define the c-subcomplex of $\mathrm{Sub}(\mathcal{I})$ as $\mathrm{Sub}(\mathcal{I})_c = \{\sigma \in \mathrm{Sub}(\mathcal{I}) \mid c \in \chi^*(\sigma)\}$.*

The following quite obvious Lemma 18 shows that independently subdivided c-subcomplexes $\mathrm{Sub}(\mathcal{I})$ can be globally refined.

▶ **Lemma 18.** *Let $\mathcal{I}_1, \ldots, \mathcal{I}_k$ be subcomplexes of $\mathcal{I}$ such that $\bigcup_{j=1}^{k} \mathcal{I}_j = \mathcal{I}$, and $\mathrm{Sub}(\mathcal{I}_j)$ be a chromatic subdivision of each $\mathcal{I}_j$. There exists a chromatic subdivision $\mathrm{Sub}(\mathcal{I})$ that refines each Sub($\mathcal{I}_j$).*

Since a given facet $\sigma \in \mathrm{Sub}(\mathcal{I})$ may be c-covered for several different colors c, we pick one of those to obtain a color partition of $\mathrm{Sub}(\mathcal{I})$, as provided by Lemma 19.

▶ **Lemma 19.** *Let $\mathrm{Sub}(\mathcal{I})$ be a star-covered subdivision with respect to $f : |\mathcal{I}| \to |\mathcal{O}|$, with its induced i-colored subcomplexes $\mathrm{Sub}(\mathcal{I})_i$, $1 \le i \le m$. There exists a partition $\mathcal{P} = \{P_1, \ldots, P_r\}$, $1 \le r \le m$, of $\mathrm{Sub}(\mathcal{I})$ such that each $P_i \subseteq \mathrm{Sub}(\mathcal{I})_i$, and for any pair $i < j$, $P_j \cap \mathrm{Sub}(\mathcal{I})_i = \varnothing$. We call $\mathcal{P}$ the color partition of $\mathrm{Sub}(\mathcal{I})$.*

Proof. Let $C_1 = \mathrm{Sub}(\mathcal{I})$, which is a star-covered subdivision with respect to f. According to Lemma 15, not all i-subcomplexes of C_1 are empty. As we can find a permutation of the coloring $\chi(\mathcal{I})$ that ensures $\mathrm{Sub}(\mathcal{I})_1 \neq \varnothing$, we can just define $P_1 = \mathrm{Sub}(\mathcal{I})_1$. Now consider $C_2 = \mathrm{Sub}(\mathcal{I}) \setminus P_1$. If $C_2 = \varnothing$, then $\mathcal{P} = \{P_1\}$. Otherwise, we can proceed inductively to define the remaining P_i, $i \ge 2$. ◀

The following Lemma 20 is instrumental in the proof of our chromatic simplicial approximation Theorem 22. It shows that chromatic projections applied to chromatic functions provide chromatic functions, as illustrated in Figure 5.

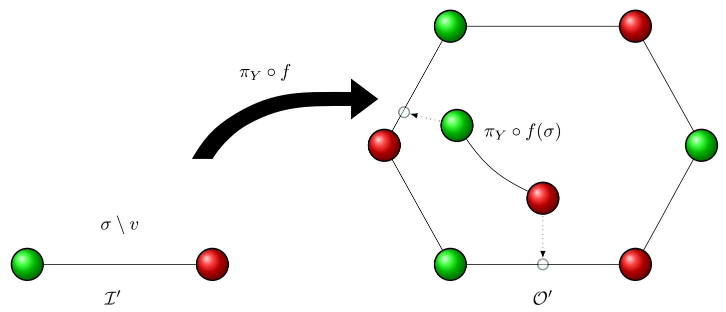


Figure 5 The chromatic projection π_Y, applied to the chromatic function shown in Figure 4 with v denoting the yellow (Y) vertex, induces a chromatic function in a lower dimension. The function $f(x)$ maps the green vertex $x \in \sigma \setminus \{v\}$, the red-green border of σ, to $f(x)$ represented by the green inner vertex, the carrier of which contains the yellow central vertex in Figure 4. Applying $\pi_c(f(x)$ retracts this point to the opposite red-green border.

▶ **Lemma 20.** *Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a chromatic function, and $\mathrm{Sub}(\mathcal{I})$ a star-covered subdivision with respect to f. Assume that S_c is a c-colored subcomplex of $\mathrm{Sub}(\mathcal{I})$, and let $S'_c = S_c \setminus \{v \in V(S_c) \mid \chi(v) = c\}$, $\mathcal{O}'_c = \mathcal{O} \setminus \{v \in V(\mathcal{O}) \mid \chi(v) = c\}$. Then, $f_c : |S'_c| \to |\mathcal{O}'_c|$ defined by $f_c(x) = \pi_c(f(x))$ is a chromatic function.*

Proof. Let $V_c = \{v \in V(\mathrm{Sub}(\mathcal{I})) \mid \chi(v) = c\}$. Notice that from Definition 13, π_c is continuous in $|S_c \setminus V_c|$. Since f is continuous, $\pi_c \circ f$ must also be continuous in $|S'_c| \subset |S_c \setminus V_c|$. Hence, $f_c(x) = \pi_c(f(x))$ is indeed continuous.

In order to show that $f_c(x)$ is chromatic, let N be an r-dimensional simplicial neighborhood in $|S'_c|$. Note that N is also an r-dimensional neighborhood in $\mathrm{Sub}(\mathcal{I})$. Corollary 7 implies that f is also chromatic in $|S'_c|$, hence $|\,|\chi|(N) \cap |\chi|(f(N))| \geq r+1$ holds. Since $c \notin |\chi|(N)$ as $N \subseteq |S'_c|$, it follows that $c \notin |\chi|(N) \cap |\chi|(f(N))$. Consequently, $|\chi|(N) \cap |\chi|(f(N)) = (|\chi|(N) \cap |\chi|(f(N))) \setminus \{c\} = |\chi|(N) \setminus \{c\} \cap |\chi|(f(N)) \setminus \{c\}$ by distributivity of set exclusion over intersection. Recalling $c \notin |\chi|(N)$ and noticing that $|\chi|(\pi_c \circ f(N)) = |\chi|(f(N)) \setminus \{c\}$, it follows that $|\chi|(N) \cap |\chi|(f(N)) = |\chi|(N) \setminus \{c\} \cap |\chi|(f(N)) \setminus \{c\} = |\chi|(N) \cap |\chi|(\pi_c \circ f(N))$. We conclude that $|\,|\chi|(N) \cap |\chi|(f_c(N))| = |\,|\chi|(N) \cap |\chi|(f(N))| \geq r+1$. Therefore, f_c is indeed a chromatic function. ◀

The following Lemma 21 is just the chromatic approximation theorem Theorem 22 written out for 1-dimensional simplicial complexes. It will serve as the induction basis for the proof of the latter.

▶ **Lemma 21.** *Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a continuous chromatic function such that $\mathcal{I}$ and $\mathcal{O}$ are pure simplicial complexes of dimension 1. There exists a chromatic subdivision $\mathrm{Sub}(\mathcal{I})$ of $\mathcal{I}$, and a chromatic simplicial map $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ that is a chromatic approximation of f.*

Proof. From Lemma 15, it follows that there exists a chromatic subdivision $\mathrm{Sub}(\mathcal{I})$ which is star-covered with respect to f. We define a simplicial vertex map $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ as follows: For a vertex v, we set $\mu(v) = w$ if $\chi(w) = \chi(v)$ and $w \in \mathrm{carr}(f(v), \mathcal{O})$. That is, μ maps vertex v to the vertex of the same color in the carrier of $f(v)$, so obviously is a chromatic map.

Now consider a facet $\sigma = \{v_1, v_2\} \in \mathrm{Sub}(\mathcal{I})$. Since f is star-covered, $f(\mathrm{int}(|\sigma|)) \subseteq \mathrm{star}^o(w_1)$ for some $w_1 \in \mathcal{O}$. Assume w.l.o.g. that $\mu(v_1) = w_1$, and let $w_2 = \mu(v_2)$. Notice that $w_2 \in \mathrm{star}(w_1)$, as otherwise $f(\mathrm{int}(|\sigma|)) \subseteq \mathrm{star}^o(w_1)$ is impossible. Therefore, there exists a simplex $\tau \in \mathcal{O}$ that includes w_2 and w_1. It follows that $\tau = \mu(\sigma) \in \mathcal{O}$, which proves that μ is a chromatic simplicial map. It follows from the definition of μ that it is a chromatic approximation to f. ◀

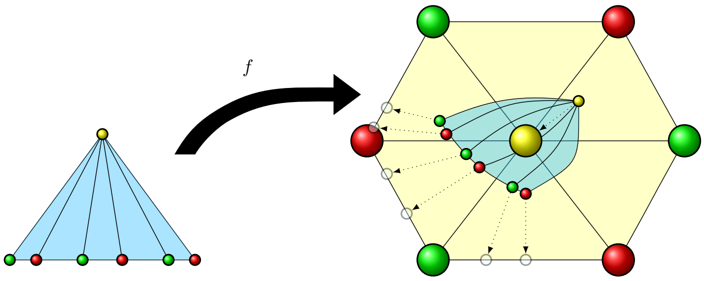


Figure 6 Star-cover for the chromatic subdivision $\mathrm{Sub}^0(\mathcal{I})$, which induces a trivial color partition consisting of a single $\mathcal{K}_1^0$, with color 1 representing yellow, containing all images of the subdivides simplices in $\mathrm{Sub}^0(\mathcal{I})$. The small yellow vertex is mapped by μ_1^0 to the yellow center of the star, the small read and green vertices will be retracted to the opposite border and appropriately mapped in the induction step in the proof of Theorem 22, and finally mapped to some border vertices as shown in Figure 7.

▶ **Theorem 22** (Chromatic Approximation Theorem). *Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a chromatic function such that $\mathcal{I}$ and $\mathcal{O}$ are pure simplicial complexes of dimension k, and with the same color set $\chi(\mathcal{I}) = \chi(\mathcal{O})$. There exists a chromatic subdivision* $\mathrm{Sub}(\mathcal{I})$ *of $\mathcal{I}$, and a chromatic simplicial map $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ that is a chromatic approximation of f.*

Proof. We use induction over $\dim(\mathcal{I}) = \dim(\mathcal{O}) = k$. The induction base $k = 1$ is provided by Lemma 21. For our induction hypothesis, we assume that Theorem 22 holds for arbitrary chromatic functions with respect to $\mathcal{I}'$ and $\mathcal{O}'$ with dimension less than $k \geq 2$. Let $f : |\mathcal{I}| \to |\mathcal{O}|$ be a chromatic function with respect to $\mathcal{I}$ and $\mathcal{O}$, where $\dim(\mathcal{I}) = \dim(\mathcal{O}) = k$ and $\chi(\mathcal{I}) = \chi(\mathcal{O}) = \{1, \ldots, k+1\}$.

It follows from Lemma 15 that there exists a chromatic subdivision $\mathrm{Sub}^0(\mathcal{I})$ of $\mathcal{I}$, which is star-covered with respect to f. Lemma 19 thus ensures that there is a color partition $\mathcal{P}^0 = \{\mathcal{K}_1^0, \ldots, \mathcal{K}_r^0\}$ of $\mathrm{Sub}^0(\mathcal{I})$. Recall that each $\mathcal{K}_i^0$ is an i-covered subcomplex of $\mathrm{Sub}^0(\mathcal{I})$. Therefore, for any $\sigma \in \mathcal{K}_i^0$, $f(\mathrm{int}(|\sigma|)) \subseteq \mathrm{star}^o(w)$ for some $w \in \mathcal{O}$ with color i. Now, iteratively for $i = 1, 2, \ldots, r$, we will perform the following two steps:

As our first step for i, we define a partial vertex map $\mu_i^0 : \mathcal{K}_i^0 \to \mathcal{O}$ for vertices with color i, as illustrated for the small yellow vertex on the boundary of $f(|\sigma|)$ in Figure 6: Let $v \in V(\mathcal{K}_i^0)$ be such that $\chi(v) = i$. Since $\mathcal{K}_i^0$ is i-covered, there exists some $w \in V(\mathcal{O})$ such that $\chi(w) = i$ and $f(v) \in \mathrm{star}^o(w)$. We choose any such w, and set $\mu_i^0(v) = w$. Note carefully that this partial map μ_i^0 defines uniquely a partial map for vertices of color i also in further chromatic subdivisions of $\mathcal{K}_i^0$: Let $\mathcal{L}_i$ be any such subdivision of $\mathcal{K}_i^0$ and $v \in V(\mathcal{L}_i)$, $\chi(v) = i$. Taking v as a point in the corresponding 0-dimensional geometric simplex in $\mathcal{L}_i$, consider $\sigma = \mathrm{carr}(v, \mathcal{K}_i^0)$. Since f is chromatic also in $|\mathcal{L}_i|$ by Corollary 7, there exists a unique vertex $v' \in \sigma$ such that $\chi(v') = i$. We can therefore consistently define $\mu_i^0(v) = \mu_i^0(v')$.

As our second step for i, we define a partial vertex map $\mu_i^1 : \mathcal{K}_i^0 \to \mathcal{O}$ for vertices with a color $\neq i$, as illustrated for the small red and green vertices on the boundary of $f(|\sigma|)$ in Figure 6. Let $\mathcal{K}_i^1 = \mathcal{K}_i^0 \setminus \{v \in V(\mathcal{K}_i^0) \mid \chi(v) = i\}$ and $\mathcal{O}_i = \mathcal{O} \setminus \{v \in V(\mathcal{O}) \mid \chi(v) = i\}$. Since f is chromatic, Lemma 20 implies that each $f_i : |\mathcal{K}_i^1| \to |\mathcal{O}_i|$ defined by $f_i(x) = \pi_i(f(x))$ is also chromatic. Notice that each $\mathcal{K}_i^1$ and $\mathcal{O}_i$ has dimension less than k. Therefore, the induction hypothesis holds, and there exist $\mu_i^1 : \mathrm{Sub}(\mathcal{K}_i^1) \to \mathcal{O}_i$ that are chromatic approximations to f_i, as illustrated in Figure 7. As before, each μ_i^1 defines a vertex map for any further chromatic subdivision $\mathcal{L}_i$ of $\mathrm{Sub}(\mathcal{K}_i^1)$ in the same way as μ_i^0 did for further subdivisions of $\mathcal{K}_i^0$.

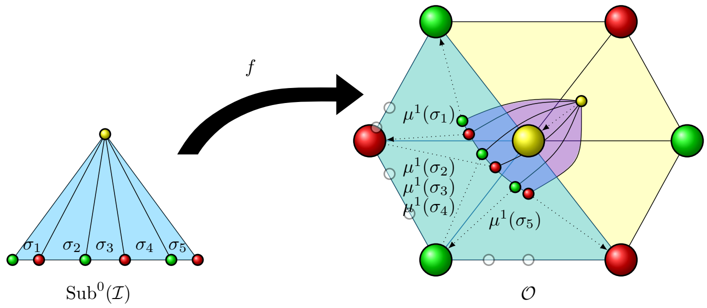


Figure 7 Mapping of the small red and green vertices in Figure 6, which were retracted to the outer border of the star, to appropriate border vertices via μ_2^1 and μ_3^1 (denoted as μ^1 in the figure for brevity) in the induction step in the proof of Theorem 22.

Since each μ_i^1 defines a vertex map for vertices of color different to i, and μ_i^0 defines a vertex map for vertices of color i, the join $\mu_i^1 * \mu_i^0 = \mu_i$ defines a vertex map for $\mathcal{K}_i^0$ for all colors.

To combine all these partial functions μ_i into a global function μ, we make use of Lemma 18. It ensures that there is a chromatic subdivision $\mathcal{K}$ of $\mathrm{Sub}(\mathcal{I})$ that refines every $\mathrm{Sub}(\mathcal{K}_i^0)$. Recall that every μ_i is well defined for any further chromatic subdivision $\mathrm{Sub}(\mathcal{K}_i^0)$. Therefore, we can globally define $\mu : \mathcal{K} \to \mathcal{O}$ as $\mu(v) = \mu_i(v)$, where i is the smallest color such that $v \in \mathrm{Sub}(\mathcal{K}_i^0)$ in a sufficiently deep chromatic subdivision of $\mathcal{K}_i^0$.

In only remains to prove that the so constructed μ is a chromatic approximation to f. It suffices to show that $\mu(\sigma) \subseteq \mathrm{carr}(f(|\sigma|), \mathcal{O})$ for every facet $\sigma = \{v_1, \dots, v_n\} \in \mathcal{K}$. Let $\mathrm{carr}(f(|\sigma|), \mathcal{O}) = \kappa = \{w_1, \dots, w_n\}$. Since it generally holds for facets in a pure simplicial complex that $\kappa = \bigcap_{i=1}^{n} \mathrm{star}(w_i)$, we only need to prove that $\mu(\sigma) \subseteq \mathrm{star}(w_i)$ for each w_i. This follows from the inductive construction of each μ_i and μ and the properties of the elements of color partition $\mathcal{P}^0$, however, which are based on star-covered subdivisions. ◀

3 Continuous Task Solvability in the ASM Model

In this section, we show that chromatic functions precisely capture the notion of computability for the wait-free asynchronous read/write shared memory (ASM) model. Our results hence indeed provide an alternative proof of the ACT Theorem 1, by means of a simple reduction based on Lemma 23.

▶ **Lemma 23.** *A continuous task $T = \langle \mathcal{I}, \mathcal{O}, f \rangle$ has a solution in ASM if and only if its induced task $T_f = \langle \mathcal{I}, \mathcal{O}, \Delta_f \rangle$ has a solution in ASM.*

Proof. Assume that a continuous task $T = \langle \mathcal{I}, \mathcal{O}, f \rangle$ has a solution, that is, that there exists a subdivision $\mathrm{Sub}(\mathcal{I})$ and a decision map $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ such that for each $\sigma \in \mathrm{Sub}(\mathcal{I}), \mu(\sigma) \subseteq \mathrm{carr}(f(|\sigma|), \mathcal{O})$. It follows from Definition 12 that μ solves T_f.

Conversely, assume that the induced task T_f has a solution. According to the ACT Theorem 1, there exists a subdivision $\mathrm{Sub}(\mathcal{I})$ and a decision map $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ carried by Δ_f. From Definition 11 of the carrier Δ_f of T_f, it follows that for any $\sigma \in \mathrm{Sub}(\mathcal{I})$, $\mu(\sigma) \subseteq \mathrm{carr}(f(|\sigma|), \mathcal{O})$. Therefore, by Definition 11, μ solves T. ◀

The following Theorem 24 establishes the equivalence of ASM task solvability and the existence of a chromatic task.

▶ **Theorem 24** (CACT). *A task $\langle \mathcal{I}, \mathcal{O}, \Delta \rangle$ is solvable in ASM if and only if there exists a continuous task $\langle \mathcal{I}, \mathcal{O}, f \rangle$ such that $f(|\sigma|) \subseteq |\Delta(\sigma)|$ for any input simplex $\sigma \in \mathcal{I}$.*

Proof. If $\langle \mathcal{I}, \mathcal{O}, \Delta \rangle$ is solvable, then there exists a subdivision $\mathrm{Sub}(\mathcal{I})$ and a decision map $\mu : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$ carried by Δ. Since μ is a chromatic simplicial map, Lemma 8 reveals that its geometric realization $|\mu| : |\mathcal{I}| \to |\mathcal{O}|$ is a chromatic function, which satisfies $|\mu|(|\sigma|) \subseteq |\Delta(\sigma)|$ for any input simplex $\sigma \in \mathcal{I}$.

Conversely, let $T = \langle \mathcal{I}, \mathcal{O}, f \rangle$ be a continuous task such that $f(|\sigma|) \subseteq |\Delta(\sigma)|$, and consider the task $T_f = \langle \mathcal{I}, \mathcal{O}, \Delta \rangle$. Since f is chromatic, it follows from Theorem 22 that f has a chromatic approximation $\mu_f : \mathrm{Sub}(\mathcal{I}) \to \mathcal{O}$. Since $f(|\sigma|) \subseteq |\Delta(\sigma)|$, Definition 11 implies that μ_f solves the induced task T_f. ◀

Finally, the following Theorem 25 shows that every continuous task can be solved in the ASM. Together with Theorem 24, it provides the continous counterpart of the only-if direction of the ACT Theorem 1.

▶ **Theorem 25.** *Any continuous task $T = \langle \mathcal{I}, \mathcal{O}, f \rangle$ is solvable in ASM.*

Proof. Let $T = \langle \mathcal{I}, \mathcal{O}, f \rangle$ be a continuous task. It follows from Theorem 22 that f has a chromatic approximation. According to Definition 11, it therefore has a solution in ASM. Lemma 23 thus confirms that the induced task $T_f = \langle \mathcal{I}, \mathcal{O}, \Delta_f \rangle$ also has solution in ASM. ◀

4 Application Example: Consensus-Preferent Approximate Agreement

As we have shown, chromatic functions precisely characterize task solvability under the wait-free asynchronous shared memory model. However, their expressive power goes way beyond that, opening up interesting future research areas. We will demonstrate this by using a continuous task to specify preferences for particular output configurations. More specifically, we consider a two-process system in ASM and specify a 1/3-binary-approximate agreement task with a fixed *preference* $0 < K < 1$ for *exact* agreement. The parameter K determines the fraction of all the executions of the system that will terminate in a configuration where all processes decide on the same value. For example, if one assumes that all IIS runs are equally likely, then K gives the probability that a randomly chosen run terminated with exact agreement.

Since every execution corresponds to a simplex in the chromatic subdivision $\mathrm{Sub}(\mathcal{I})$ guaranteed by Theorem 22, this preference constraint can be easily expressed by means of piecewise linear functions in a continuous task specification. More specifically, we will define a chromatic function $f : |\mathcal{I}| \to |\mathcal{O}|$ for the task at hand as follows (see Figure 8 for an illustration).

Let σ_0 be the simplex of $\mathcal{I}$ that corresponds to the input configuration $\{(p_1, 0)(p_2, 0)\}$, and τ_0 the simplex of $\mathcal{O}$ that corresponds to the output configuration $\{(p_1, 0), (p_2, 0)\}$. For $x \in |\sigma_0| = \lambda \cdot (p_1, 0) + (1-\lambda) \cdot (p_2, 0)$ for $0 \le \lambda \le 1$, we define $f(x) = \lambda \cdot (p_1, 0) + (1-\lambda) \cdot (p_2, 0) \in |\mathcal{O}|$. We extend this definition to $\sigma_1 = \{(p_1, 1), (p_2, 1)\} \in \mathcal{I}$ and $\tau_1 = \{(p_1, 1), (p_2, 1)\} \in \mathcal{O}$ in a completely analogous way. This corresponds to mapping the top resp. bottom edge of the input complex to the top resp. bottom edge of the output complex.

In order to complete our definition of f, we also need to define it on $\sigma_2 = \{(p_1, 0), (p_2, 1)\}$ and on $\sigma_3 = \{(p_1, 1), (p_2, 0)\}$. We describe our construction for f restricted to $|\sigma_2|$; the construction for f on $|\sigma_3|$ is completely analogous. Let M_1 be any positive number such that $0 < K < M_1 < 1$, and set $R = (1 - M_1)/3$. Consider $x_1, \ldots, x_4 \in |\sigma_2|$ defined by $x_1 = R \cdot (p_1, 0) + (M_1 + 2R) \cdot (p_2, 1)$, $x_2 = (M_1/2 + R) \cdot (p_1, 0) + (2R + M_1/2) \cdot (p_2, 1)$,

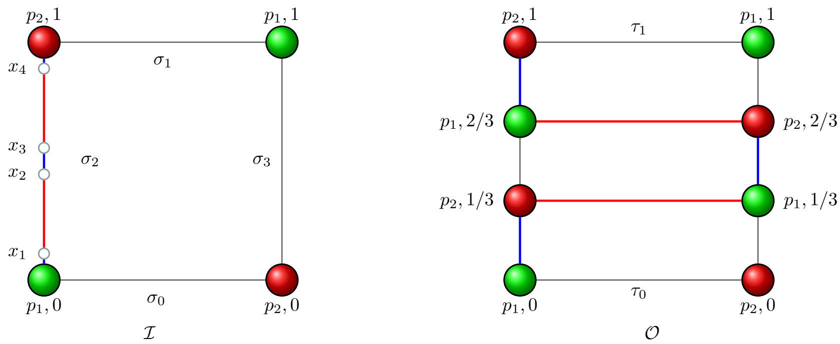


Figure 8 The chromatic function for the consensus-preferent 1/3-approximate agreement task. It maps a portion of K of the left edge in the input simplex $\mathcal{I}$, represented by the red segments (x_1, x_2) and (x_3, x_4), to the red horizontal edges in the middle of $\mathcal{O}$, and the remaining blue segments to the blue vertical edges on the left and right of $\mathcal{O}$.

$x_3 = (2R + M_1/2) \cdot (p_1, 0) + (M_1/2 + R)(p_2, 1)$, and $x_4 = (2R + M_1) \cdot (p_1, 0) + R \cdot (p_2, 1)$. We now define f on $|\sigma_2|$ as a piecewise affine function that maps segment $[(p_1, 0), x_1]$ to segment $[(p_1, 0), (p_2, 1/3)]$. More specifically, for x in segment $[(p_1, 0), x_1]$ given by $x = \lambda \cdot (p_1, 0) + (1 - \lambda) \cdot x_1$ for some $0 \leq \lambda \leq 1$, we define $f(x) = \lambda \cdot (p_1, 0) + (1 - \lambda) \cdot (p_2, 1/3)$. In the same way, we can map segments $[x_1, x_2]$ to $[(p_2, 1/3), (p_1, 1/3)]$, $[x_2, x_3]$ to $[(p_1, 1/3), (p_2, 2/3)]$, $[x_3, x_4]$ to $[(p_2, 2/3), (p_1, 2/3)]$ and $[x_4, (p_2, 1)]$ to $[(p_1, 2/3), (p_2, 1)]$.

Since it is obvious that the so-constructed f is a chromatic function, Theorem 24 guarantees that this task is solvable in the wait-free ASM model. More specifically, we obtain the following Theorem 26:

▶ **Theorem 26.** *Let Sub($\mathcal{I}$) be a chromatic subdivision of $\mathcal{I}$, such that each facet has the same size less than $(M_1 - K)/4$, and let μ be the chromatic approximation to f with respect to* $\mathrm{Sub}(\mathcal{I})$ *guaranteed by Theorem 22. Then, μ can be used to solve 1/3-approximate agreement with a proportion of at least K executions that output the same value in the wait-free ASM. In the IIS model, $r = \log_3(4/(M_1 - K))$ communication rounds are enough for solving this task.*

Proof. From the definition of f, it is apparent that the segments $[x_1, x_2]$ and $[x_3, x_4]$ both have length $M/2$. Since they are mapped to the red segments in in $\mathcal{O}$ in Figure 8, it follows that if $\sigma \in \mathrm{Sub}(\mathcal{I})$, and $|\sigma| \subseteq \mathrm{int}([x_1, x_2])$, then $\mu(\sigma) = (p_1, 1/3), (p_2, 1/3)$, and if $|\sigma| \subseteq \mathrm{int}([x_3, x_4])$, then $\mu(\sigma) = (p_1, 2/3), (p_2, 2/3)$. Since $[x_1, x_2]$ has only two border points, it must have at least $\lceil (M_1/2)/D \rceil - 2$ internal facets, where D is the facet size. Analogously, $[x_3, x_3]$ must also have at least $\lceil (M_1/2)/D \rceil - 2$ internal facets. In total, we have at least $M_1/D - 4$ facets in the original σ_2 that map to an exact consensus output configuration, out of a total of $1/D$ facets. Thus, the proportion of the subdivided simplices that results in exact agreement is at least $M_1 - 4 \cdot D$. From the statement of our theorem, we know that $D \leq (M_1 - K)/4$, therefore the proportion is at least $M_1 - (M_1 - K) = K$.

Exactly the same reasoning applies to σ_3, which also guarantees exact agreement for a proportion at least K. On the other hand, all output configurations reachable from σ_0 and σ_1 guarantee consensus, therefore their proportion is 1. Since $\sigma_0, \sigma_1, \sigma_2$, and σ_3 are all the facets of $\mathcal{I}$, and since the subdivision is uniform, it follows that the proportion of the facets of $\mathrm{Sub}(\mathcal{I})$ leading to exact consensus is at least K.

Since the standard chromatic subdivision divides a 2-dimensional simplex uniformly and reduces its size to 1/3 of the original size, Theorem 26 implies that $r = \log_3(4/(M_1 - K))$ communication rounds are indeed enough for solving this task in the IIS model. ◀

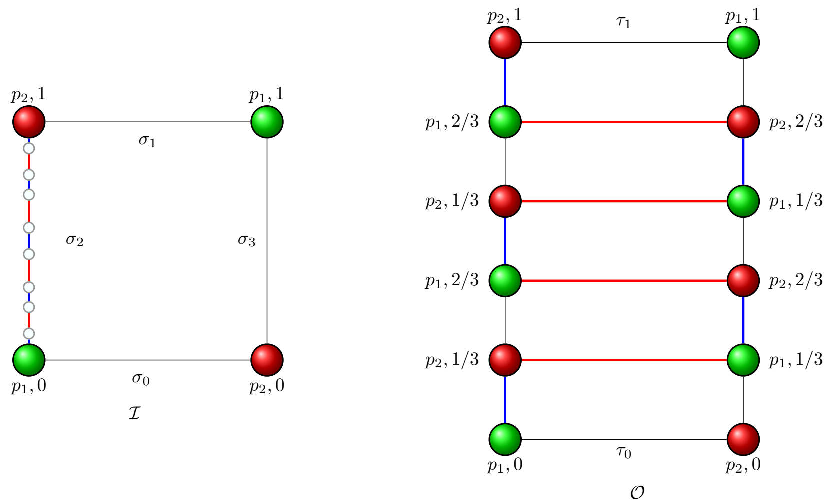


Figure 9 The chromatic function for and extended version of the consensus-preferent 1/3-approximate agreement task. In contrast to Figure 8, it allows both processes to decide on 1/3 resp. 2/3 when the other process decides on 2/3 resp. 1/3.

We conclude this section by noting that the above example also reveals that continuous tasks allow for more natural fine-grained specifications than standard tasks: It is apparent from Figure 8 that the continuous task specification does not allow a configuration where (p_1, p_2) have decided $(2/3, 1/3)$ when starting from the input $(0, 1)$ in σ_2, which would be allowed by the standard (colorless) task specification of 1/3-approximate agreement. To also allow this behavior, one could use the continuous task specification illustrated in Figure 9, however.

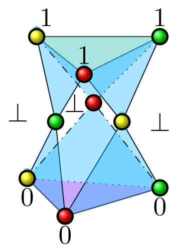
5 Discussion of Related Work

5.1 Alternative characterization approaches

In [36], Saraph, Herlihy and Gafni explain that *"... the ACT contains one difficult step. Using the classical simplicial approximation theorem, it is straightforward to construct a simplicial map having all desired properties except that of being color-preserving. To make this map color-preserving required a rather long construction employing mechanisms from point-set topology, such as balls and Cauchy sequences."* In their paper, they give an alternative proof strategy for the ACT, in which the essential chromatic property is guaranteed by a distributed convergence algorithm, rather than by a combinatorial construction, whose proof is quite subtle and long. They observe that the convergence algorithm can be applied to more general continuous functions from $\mathcal{I}$ to $\mathcal{O}$, carried by Δ, given the assumption that $\Delta(\sigma)$ is link-connected for all σ in $\mathcal{I}$. This implies [36, Theorem 6.1], which states that if there is a continuous map $f : |\mathcal{I}| \to |\mathcal{O}|$ that is carried by a link-connected carrier map Δ, then there exists a carrier-preserving simplicial map from a chromatic subdivision of $\mathcal{I}$ to $\mathcal{O}$. We note, however, that this is only a sufficient condition, but nothing is stated about the other direction: Whereas constructing a continuous map from a carrier-preserving simplicial map is easy, it is not clear how to enforce link-connectivity of Δ.

In fact, link-connectivity is a property required from the output complex resp. the images of the carrier map Δ, and it is easy to find examples (like the Hourglass task discussed in Section 1) of continuous tasks that are not link-connected. In particular, the assumption that $\Delta(\sigma)$ is link-connected can be violated even by admitting just one additional possible output configuration $\rho \in \Delta(\sigma)$ (e.g., involving a default value $\perp$ as in quasi-consensus [19]) for just one simplex $\sigma \in \mathcal{I}$.

We illustrate this in more detail by means of a general *k-failsafe* extension, which could be added to any conventional task specification: Rather than deciding on some output value, any process, up to a maximum of k processes, is also allowed to abstain. Abstaining can be formalized by outputting a distinguished value denoted by $\perp$. If $\mathcal{T}$ is a non-trivial task and $k < n-1$, then the output complex of the k-failsafe version of T is not link-connected. This follows from the fact that any face F containing k yielding processes must be connected to any face consisting of the complementary $n-k > 1$ non-yielding processes. Removing F would lead to disconnected, i.e., (-1)-connected, facets of the original task, whereas link-connectivity would require a connectivity of $n-1-(k-1)-2 > -1$. For example, the output complex of 1-failsafe consensus with 3 processes shown in the figure below is not link-connected; note that it can be viewed as three intertwined instances of the Hourglass task. In sharp contrast to link-connectivity-based characterizations, our continuous task-based one can be applied here, and reveals that the task cannot be solved in wait-free ASM.



We finally note that whereas it could be argued that one could try to restrict a non-link-connected output complex resp. carrier map to a link-connected one, this is not always the case: there are examples like the valency task introduced in [4], where any such restriction would render an unsolvable task solvable.

The only alternative/generalized characterization that we are aware of can be found in [16], however it requires the output complex resp. the carrier map to be link-connected. In particular, Gafni, Kuznetsov and Manolescu provide a generalized ACT for general Sub-IIS models, leading to affine tasks [30], which also include non-compact models like t-resilient and obstruction-free ones. Their GACT depends on the existence of a chromatic simplicial map, however. In the wait-free case [16, Cor. 7.1], the authors just refer to the construction in [28]. For general models, a dedicated [16, Thm. 8.4] is provided. However, it requires a link-connected output complex and just resorts to [28, Lem. 4.21, Thm. 5.29] in its proof as well.

A related alternative approach uses distributed computing arguments to limit the type of chromatic subdivisions that take place, to avoid the need for the difficult perturbation arguments of the original ACT proof. Such an approach is described in [8], which uses a simulation [17] to the iterated model using immediate snapshot tasks.

Once a characterization has been proved for a specific distributed computing model, one may use algorithmic simulations and reductions to other models. For instance, once the ACT has been used to characterize wait-free task solvability, the BG simulation [9] can be used to characterize solvability when at most t processes may crash, by a distributed simulation of an algorithm that solves the task in the other model. Whereas this simulation works only for colorless tasks (see below), there has also been a proposal for an extension of the

BG simulation that works for general tasks [15]. Another example is [31], which presents a characterization of task computability in the wait-free shared-memory model in which processes, in addition to read/write registers, have access to k-test-and-set objects. Instead of algorithmic simulations, it is also possible to directly construct reductions (maps) from a protocol complex in one model to a protocol complex in another model [25].

5.2 Colorless computability

There is a continuous characterization of task solvability for *colorless tasks* [9], in which we care only about the sets of input and output values, but not which processes are associated with which values. Many of the main tasks of interest in distributed computing are colorless: consensus, set agreement, approximate agreement, loop agreement, etc. These tasks are defined by the colorless input complex $\mathcal{I}$, the colorless output complex $\mathcal{O}$, and an input/output relation Δ. Furthermore, colorless tasks can be solved by simpler *colorless algorithms*, where the processes consider the values read from the shared memory as a set, disregarding which value belongs to which process.

The colorless ACT [27, Theorem 8] or [22, Section 4.3] roughly says that a colorless task is wait-free solvable if and only if there is a continuous map f from $|\mathcal{I}|$ to $|\mathcal{O}|$ carried by Δ, where $|\cdot|$ denotes the geometric realization of the respective complex. Or equivalently, if and only if there is some r and a simplicial map δ from $\mathrm{Bary}^r(\mathcal{I})$ to $\mathcal{O}$ carried by Δ. The proof of the colorless ACT does not require the technicalities of the original ACT result; it uses only the classic simplicial approximation theorem.

Overall, colorless computability is fairly well understood, because the topological techniques are simpler; the whole first part of the book [22] is devoted to colorless tasks. Remarkably, some of the most important wait-free results like the set agreement impossibility and the general task undecidability [14, 23] do not require the chromatic version. Moreover, there are colorless task solvability characterizations for various other read/write models of computation, such as t-resilience [35], closed adversaries [26] and fair adversaries [32], even in anonymous systems where processes have no ids [11] and for robot coordination algorithms [2].

It would be interesting to extend existing colorless results to general tasks via our CACT, in particular, for dependent failures [24, Theorem 4.3] and [26], as well as for systems admitting solo executions [27] or partitions [18]. For solo executions, a solvability characterization for arbitrary tasks exists [37], but it is not continuous.

5.3 ASM Models

One of the central challenges in the theory of distributed computing is determining the computability power of its numerous models, parameterized by types of failures (crash, omission, Byzantine), synchrony (asynchronous, partially synchronous, synchronous), and communication primitives (message-passing, read-write registers, powerful shared objects). For various asynchronous, synchronous and semisynchronous models, the topology of the protocol complex is known [22, Chapter 13], but not a full characterization in the ACT style.

The most basic models (which preserve the topology of the input complex) are the ASM models, but even here, there are many variants that are all equivalent with respect to task solvability. The original ACT result was developed in the specific model of read/write shared memory, and was redone later in the simpler to analyze IIS model. This resulted in another, discrete version of the ACT, justified by the existence of simulations between both models [17]. The book [22, Section 14.2] describes five natural models of asynchronous computation and shows they are all equivalent with respect to task solvability. The variants are based on whether processes can take snapshots of the shared memory or just read individual registers one at a time, and whether the shared memory can be read and written multiple times, or just once (iterated models). Whereas not all ASM models have exactly the same protocol complex, they have a protocol complex that is collapsible, see e.g. [6].

As mentioned in the introduction, instead of working in a specific read/write wait-free model, we simply assume a model where Theorem 1 holds, i.e., any ASM model. Our continuous characterization can hence be viewed as subsuming the various discrete characterizations.

Another class of models where communication is by message passing is called *dynamic networks*. Here reliable processes run in synchronous rounds, an adversary defines the possible patterns of message losses. Some adversaries define models that are ASM, and hence our results apply. An investigation of the minimum set of messages whose delivery must be guaranteed to ensure the equivalence of this model to asynchronous wait-free shared-memory is presented in [1]. For general adversaries, a characterization is known only for consensus [33].

6 Conclusions and Future Work

In this paper, we defined chromatic functions as continuous functions with some color preservation properties, and showed that they precisely characterize task solvability in a wait-free asynchronous shared memory model (ASM). Chromatic functions can be seen as the continuous analogue of chromatic simplicial maps, and provide a means for expressing refined task specifications, e.g., for resolving non-determinism. Overall, they provide a purely topological formalization of the computability power of ASM models.

Technically, our results rest on the novel notion of chromatic approximations, a chromatic analog of the well-known simplicial approximation theorem. The main feature guaranteed by chromatic functions is preserving the color structure, which is not native to point-set topology. In fact, chromatic functions are a formalization of quite intuitive rules of what can and can't be done to transform an input complex into an output complex: For example, stretching and even possibly folding over along an edge or a facet is permitted. Puncturing or cutting would violate continuity and is hence forbidden, however. Less obvious is the fact that any collapsing is prohibited, since this would imply a violation to the color structure.

The utility of continuous tasks is not limited to standard task solvability. We demonstrated this fact by adding density constraints for the outputs to the classic 1/3-approximate agreement task, namely, a lower bound on the fraction of executions that actually guarantee exact consensus. This gives a guarantee that exact agreement will happen frequently enough on average.

Regarding future work, our results open up promising research avenues in several directions. Our perspective might be useful to derive characterizations of task solvability in other models of computation, especially those for which there exist no topological characterization; one example are models where communication is performed via shared objects used in practice, which are more powerful than read/write operations. Besides exploring the power and limitations (in particular, decidability in different models of computation) of continuous tasks for refining task specifications, we also believe that the ability to incorporate density constraints is interesting from a non-worst-case quality-of-service perspective in general. Last but not least, we expect that continuous tasks with density constraints could also be useful for characterizing non-terminating tasks, such as asymptotic and approximate consensus [13] and long-lived approximate agreement, which was originally presented by Herlihy in [21] and later made more precise by Attiya et al. in [3]. This is because task solvability is expressed only in terms of geometric simplicial complexes, which may allow to incorporate output sequences via continuous functions defined on the output complex.

References

1 Yehuda Afek and Eli Gafni. A simple characterization of asynchronous computations. *Theoretical Computer Science*, 561:88–95, 2015. Special Issue on Distributed Computing and Networking. doi:10.1016/j.tcs.2014.07.022.

2 Manuel Alcantara, Armando Castañeda, David Flores-Peñaloza, and Sergio Rajsbaum. The topology of look-compute-move robot wait-free algorithms with hard termination. *Distributed Comput.*, 32(3):235–255, 2019. doi:10.1007/s00446-018-0345-3.

3 Hagit Attiya, Armando Castañeda, Danny Hendler, and Matthieu Perrin. Separating lock-freedom from wait-freedom. In Calvin Newport and Idit Keidar, editors, *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing, PODC 2018, Egham, United Kingdom, July 23-27, 2018*, pages 41–50. ACM, 2018. doi:10.1145/3212734.3212739.

4 Hagit Attiya, Armando Castañeda, and Sergio Rajsbaum. Locally Solvable Tasks and the Limitations of Valency Arguments. In Quentin Bramas, Rotem Oshman, and Paolo Romano, editors, *24th International Conference on Principles of Distributed Systems (OPODIS 2020)*, volume 184 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 18:1–18:16, Dagstuhl, Germany, 2021. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.OPODIS.2020.18.

5 Hagit Attiya and Sergio Rajsbaum. The combinatorial structure of wait-free solvable tasks. *SIAM J. Comput.*, 31(4):1286–1313, 2002. doi:10.1137/S0097539797330689.

6 Fernando Benavides and Sergio Rajsbaum. Collapsibility of read/write models using discrete morse theory. *J. Appl. Comput. Topol.*, 1(3-4):365–396, 2018. doi:10.1007/s41468-018-0011-7.

7 Elizabeth Borowsky and Eli Gafni. Generalized FLP impossibility result for t-resilient asynchronous computations. In *STOC '93: Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, pages 91–100, New York, NY, USA, 1993. ACM. doi:10.1145/167088.167119.

8 Elizabeth Borowsky and Eli Gafni. A simple algorithmically reasoned characterization of wait-free computation (extended abstract). In *Proceedings of the Sixteenth Annual ACM Symposium on Principles of Distributed Computing*, PODC '97, page 189?198, New York, NY, USA, 1997. Association for Computing Machinery. doi:10.1145/259380.259439.

9 Elizabeth Borowsky, Eli Gafni, Nancy Lynch, and Sergio Rajsbaum. The BG distributed simulation algorithm. *Distributed Computing*, 14(3):127–146, 2001. URL: http://link.springer.de/link/service/journals/00446/bibs/1014003/10140127.htm.

10 Armando Castañeda and S. Rajsbaum. New combinatorial topology bounds for renaming: the lower bound. *Distributed Computing*, 22:287–301, 2010.

11 Carole Delporte-Gallet, Hugues Fauconnier, Sergio Rajsbaum, and Nayuta Yanagisawa. A characterization of t-resilient colorless task anonymous solvability. In Zvi Lotker and Boaz Patt-Shamir, editors, *Structural Information and Communication Complexity - 25th International Colloquium, SIROCCO 2018, Ma'ale HaHamisha, Israel, June 18-21, 2018, Revised Selected Papers*, volume 11085 of *Lecture Notes in Computer Science*, pages 178–192. Springer, 2018. doi:10.1007/978-3-030-01325-7_18.

12 Michael J. Fischer, Nancy A. Lynch, and M. S. Paterson. Impossibility of distributed consensus with one faulty process. *Journal of the ACM*, 32(2):374–382, April 1985.

13 Matthias Függer, Thomas Nowak, and Manfred Schwarz. Tight bounds for asymptotic and approximate consensus. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*, PODC '18, pages 325–334, New York, NY, USA, 2018. ACM. doi:10.1145/3212734.3212762.

14 E. Gafni and E. Koutsoupias. Three-processor tasks are undecidable. *SIAM J. Comput.*, 28:970–983, 1999.

15 Eli Gafni. The extended bg-simulation and the characterization of t-resiliency. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, pages 85–92, New York, NY, USA, 2009. Association for Computing Machinery. doi:10.1145/1536414.1536428.

16 Eli Gafni, Petr Kuznetsov, and Ciprian Manolescu. A generalized asynchronous computability theorem. In *Proceedings of the 2014 ACM Symposium on Principles of Distributed Computing*, PODC '14, pages 222–231, New York, NY, USA, 2014. Association for Computing Machinery. `doi:10.1145/2611462.2611477`.

17 Eli Gafni and Sergio Rajsbaum. Distributed programming with tasks. In Chenyang Lu, Toshimitsu Masuzawa, and Mohamed Mosbah, editors, *Principles of Distributed Systems - 14th International Conference, OPODIS 2010, Tozeur, Tunisia, December 14-17, 2010. Proceedings*, volume 6490 of *Lecture Notes in Computer Science*, pages 205–218. Springer, 2010. `doi:10.1007/978-3-642-17653-1_17`.

18 Hugo Rincon Galeana, Kyrill Winkler, Ulrich Schmid, and Sergio Rajsbaum. A topological view of partitioning arguments: Reducing k-set agreement to consensus. In *Stabilization, Safety, and Security of Distributed Systems - 21st International Symposium, SSS 2019, Pisa, Italy, October 22-25, 2019, Proceedings*, volume 11914 of *Lecture Notes in Computer Science*, pages 307–322. Springer, 2019. `doi:10.1007/978-3-030-34992-9_25`.

19 Rachid Guerraoui and Michel Raynal. The Alpha of Indulgent Consensus. *The Computer Journal*, 50(1):53–67, August 2006. `doi:10.1093/comjnl/bxl046`.

20 Michiel Hazewinkel. Encyclopedia of mathematics, 2018. URL: `http://encyclopediaofmath.org/index.php?title=Simplicial_complex&oldid=42757`.

21 Maurice Herlihy. Impossibility results for asynchronous PRAM (extended abstract). In *Proceedings of the Third Annual ACM Symposium on Parallel Algorithms and Architectures*, SPAA '91, pages 327–336, New York, NY, USA, 1991. Association for Computing Machinery. `doi:10.1145/113379.113409`.

22 Maurice Herlihy, Dmitry N. Kozlov, and Sergio Rajsbaum. *Distributed Computing Through Combinatorial Topology*. Morgan Kaufmann, 2013. URL: `https://store.elsevier.com/product.jsp?isbn=9780124045781`.

23 Maurice Herlihy and Sergio Rajsbaum. The decidability of distributed decision tasks (extended abstract). In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, STOC '97, pages 589–598, New York, NY, USA, 1997. Association for Computing Machinery. `doi:10.1145/258533.258652`.

24 Maurice Herlihy and Sergio Rajsbaum. The topology of shared-memory adversaries. In Andréa W. Richa and Rachid Guerraoui, editors, *Proceedings of the 29th Annual ACM Symposium on Principles of Distributed Computing, PODC 2010, Zurich, Switzerland, July 25-28, 2010*, pages 105–113. ACM, 2010. `doi:10.1145/1835698.1835724`.

25 Maurice Herlihy and Sergio Rajsbaum. Simulations and reductions for colorless tasks. In Darek Kowalski and Alessandro Panconesi, editors, *ACM Symposium on Principles of Distributed Computing, PODC '12, Funchal, Madeira, Portugal, July 16-18, 2012*, pages 253–260. ACM, 2012. `doi:10.1145/2332432.2332483`.

26 Maurice Herlihy and Sergio Rajsbaum. The topology of distributed adversaries. *Distributed Comput.*, 26(3):173–192, 2013. `doi:10.1007/s00446-013-0189-9`.

27 Maurice Herlihy, Sergio Rajsbaum, Michel Raynal, and Julien Stainer. From wait-free to arbitrary concurrent solo executions in colorless distributed computing. *Theoretical Computer Science*, 683:1–21, 2017. `doi:10.1016/j.tcs.2017.04.007`.

28 Maurice Herlihy and Nir Shavit. The topological structure of asynchronous computability. *J. ACM*, 46(6):858–923, November 1999. Conference version in ACM STOC 1993. `doi:10.1145/331524.331529`.

29 Gunnar Hoest and Nir Shavit. Toward a topological characterization of asynchronous complexity. *SIAM J. Comput.*, 36(2):457–497, 2006. `doi:10.1137/S0097539701397412`.

30 Petr Kuznetsov. Understanding non-uniform failure models. *Bull. EATCS*, 106:53–77, 2012. URL: `http://eatcs.org/beatcs/index.php/beatcs/article/view/80`.

31 Petr Kuznetsov and Thibault Rieutord. Affine tasks for k-test-and-set. In Stéphane Devismes and Neeraj Mittal, editors, *Stabilization, Safety, and Security of Distributed Systems*, pages 151–166, Cham, 2020. Springer International Publishing.

32 Petr Kuznetsov, Thibault Rieutord, and Yuan He. An asynchronous computability theorem for fair adversaries. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing, PODC 2018, Egham, United Kingdom, July 23-27, 2018*, pages 387–396, 2018. URL: `https://dl.acm.org/citation.cfm?id=3212765`.

33 Thomas Nowak, Ulrich Schmid, and Kyrill Winkler. Topological characterization of consensus under general message adversaries. In *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing, PODC 2019, Toronto, ON, Canada, July 29 - August 2, 2019.*, pages 218–227, 2019. (full version: http://arxiv.org/abs/1905.09590). `doi:10.1145/3293611.3331624`.

34 Michael Saks and Fotios Zaharoglou. Wait-free k-set agreement is impossible: The topology of public knowledge. *SIAM J. Comput.*, 29(5):1449–1483, 2000. `doi:10.1137/S0097539796307698`.

35 Vikram Saraph, Maurice Herlihy, and Eli Gafni. Asynchronous computability theorems for t-resilient systems. In Cyril Gavoille and David Ilcinkas, editors, *Distributed Computing - 30th International Symposium, DISC 2016, Paris, France, September 27-29, 2016. Proceedings*, volume 9888 of *Lecture Notes in Computer Science*, pages 428–441. Springer, 2016. `doi:10.1007/978-3-662-53426-7_31`.

36 Vikram Saraph, Maurice Herlihy, and Eli Gafni. An algorithmic approach to the asynchronous computability theorem. *J. Appl. Comput. Topol.*, 1(3-4):451–474, 2018. `doi:10.1007/s41468-018-0014-4`.

37 Yunguang Yue, Fengchun Lei, Xingwu Liu, and Jie Wu. Asynchronous computability theorem in arbitrary solo models. *Mathematics*, 8(5), 2020. `doi:10.3390/math8050757`.

A Combinatorial Topology in Distributed Computing Preliminaries

For ease of reference, we provide a collection of basic combinatorial topology definitions and their application in distributed computing, which have primarily been taken from [22]: Appendix A.1 is dedicated to basic combinatorial topology, Appendix A.2 adds definitions required for the modeling of a distributed systems using combinatorial topology.

A.1 Combinatorial topology

The first definition that we need is the definition of an abstract simplicial complex, which can can be thought of as a "high-dimensional" graph. Abstract simplicial complexes have the advantage of having a discrete combinatorial construction, while at the same time, they are able to express triangulated manifolds.

▶ **Definition 27** (Abstract simplicial complex). *An abstract simplicial complex $\mathcal{K}$ is a pair $\langle V(\mathcal{K}), F(\mathcal{K})\rangle$, where $V(\mathcal{K})$ is a set, $F(\mathcal{K}) \subseteq 2^{V(\mathcal{K})}$, and for any $\sigma, \tau \in 2^{V(\mathcal{K})}$ such that $\sigma \subseteq \tau$ and $\tau \in F(\mathcal{K})$, then $\sigma \in F(\mathcal{K})$. $V(\mathcal{K})$ is called the vertex set, and $F(\mathcal{K})$ is the face set of $\mathcal{K}$. The elements of $V(\mathcal{K})$ are called the vertices, and the elements of $F(\mathcal{K})$ are called the faces or simplices. We say that an abstract simplicial complex is* finite *if its vertex set is finite.*

We say that a simplex σ is a facet *if it is maximal with respect to containment.*

Traditionally, dimension is a number that represents how many axes we need to describe a point in space. The dimension of a simplex of an abstract simplicial complex is just the number of its vertices minus 1.

▶ **Definition 28** (Dimension). *Let $\mathcal{K}$ be an abstract simplicial complex, and $\sigma \in F(\mathcal{K})$ be a simplex. We say that σ has dimension k, denoted by $dim(\sigma) = k$, if it has a cardinality of $k + 1$. We say that $\mathcal{K}$ is of dimension k if it has a simplex of maximum dimension k.*

In the context of distributed computing, local processes' states correspond to vertices, and global configurations correspond to faces. Consequently, we only consider simplicial complexes where all simplices are of the same dimension. We call this particular type of simplicial complexes *pure abstract simplicial complexes.*

Now that we have defined the basic objects in combinatorial topology, we can define the morphisms that preserve the structure.

▶ **Definition 29** (Simplicial maps). *Let $\mathcal{K}$ and $\mathcal{L}$ be abstract simplicial complexes. We say that a vertex map $\mu : V(\mathcal{K}) \to V(\mathcal{L})$ is a simplicial map if for any $\sigma \in F(\mathcal{K}), \mu(\sigma) \in \mathcal{L}$.*

We say that a simplicial map μ is rigid *if for any σ of dimension d, $\mu(\sigma)$ is also of dimension d.*

It should be noted that simplicial maps are the discrete analogon to continuous functions. This equivalence is formally stated through the simplicial approximation theorem.

In the context of distributed systems, it is sometimes inevitable to enrich simplicial complexes with a coloring. Such a coloring corresponds to extracting the processes' ids from the local states.

▶ **Definition 30** (Coloring). *Let $\mathcal{K}$ be a finite abstract simplicial complex of dimension k. We say that a function $\chi : V(\mathcal{K}) \to \{1, 2, \ldots, k+1\}$ is a proper coloring if for any simplex $\sigma \in F(\mathcal{K})$, χ is injective at σ.*

In order to simplify notation, whenever we have two disjoint abstract simplicial complexes $\mathcal{K}$ and $\mathcal{L}$, both of dimension k, with colorings μ_1 and μ_2, we will instead implicitly consider a "global" coloring $\mu : V(\mathcal{K}) \cup V(\mathcal{L}) \to \{1, \ldots, k+1\}$, since it is simpler and usually free from ambiguity.

Since we are interested that morphisms preserve the color structure, we must also add a coloring restriction.

▶ **Definition 31** (Chromatic simplicial maps). *Let $\mathcal{K}$ resp. $\mathcal{L}$ be abstract simplicial complexes of dimension k, and χ_1 resp. χ_2 proper colorings for $\mathcal{K}$ resp. $\mathcal{L}$. We say that a simplicial map is chromatic if for any $v \in V(\mathcal{K})$, $\chi_1(v) = \chi_2(\mu(v))$.*

It should be noted that all chromatic simplicial maps are rigid.

We need to define a substructure relation. Notice that since an abstract simplicial complex is a pair of two sets, it is natural to define the substructure relation based on set containment.

▶ **Definition 32** (Subcomplex). *Let $\mathcal{K}$ and $\mathcal{L}$ be abstract simplicial complexes, we say that $\mathcal{L}$ is a subcomplex of $\mathcal{K}$ if $V(\mathcal{L}) \subseteq V(\mathcal{K})$ and $F(\mathcal{L}) \subseteq F(\mathcal{K})$.*

Since we have a definition for the subcomplex relation, we can now define the k-dimensional skeleton of an abstract simplicial complex.

▶ **Definition 33** (r-Skeleton). *Let $\mathcal{I}$ be a k-dimensional abstract simplicial complex, and $r \leq k$. We define the r-skeleton of I, Skel-r$(\mathcal{I})$ as the subcomplex induced by the r-dimensional simplices of $\mathcal{I}$. More precisely:*

$$V(\textit{Skel-r}(\mathcal{I})) = \{v \in V(\mathcal{I}) \mid \exists\, \sigma \in F(\mathcal{I}) \wedge \textit{dim}(\sigma) = r \wedge v \in \sigma\},$$
$$F(\textit{Skel-r}(\mathcal{I})) = \{\sigma \in F(\mathcal{I}) \mid \exists\, \tau \in F(\mathcal{I}) \wedge \textit{dim}(\tau) = r \wedge \sigma \subseteq \tau\}.$$

▶ **Definition 34** (Carrier Map). *Let $\mathcal{K}$ and $\mathcal{L}$ be abstract simplicial complexes and $\Phi : \mathcal{K} \to 2^{\mathcal{L}}$. We say that Φ is a carrier map if $\Phi(\sigma)$ is a subcomplex of $\mathcal{L}$ for any $\sigma \in \mathcal{K}$, and for any $\sigma_1, \sigma_2 \in \mathcal{L}$, $\Phi(\sigma_1 \cap \sigma_2) \subseteq \Phi(\sigma_1) \cap \Phi(\sigma_2)$.*

We say that a carrier map $\Phi : \mathcal{K} \to 2^{\mathcal{L}}$ is rigid *if for every simplex $\sigma \in \mathcal{K}$ of dimension d, the subcomplex $\Phi(\sigma)$ is pure of dimension d. It is is* strict *if for every simplices $\sigma, \tau \in \mathcal{K}$, $\Phi(\sigma \cap \tau) = \Phi(\sigma) \cap \Phi(\tau)$. Finally, it carries a simplicial vertex map $\mu : \mathcal{K} \to \mathcal{L}$ if, for any $\sigma \in \mathcal{K}$, $\mu(\sigma) \in \Phi(\sigma)$.*

▶ **Definition 35** (Star of a vertex)**.** *Let $v \in V(\mathcal{K})$. We define the star of v as the subcomplex of $\mathcal{K}$ of all simplices that contain v.*

Subdivisions are used to create refined simplicial complexes out of a given simplicial complex. In the barycentric subdivision, the original faces become vertices, and simplex chains become the faces of the new subdivision.

▶ **Definition 36** (Barycentric subdivision)**.** *Let $\mathcal{K}$ be an abstract simplicial complex. We define the barycentric subdivision, denoted by $Bary(\mathcal{K})$ through its vertex set and faces as follows.*

$$V(Bary(\mathcal{K})) = F(\mathcal{K}),$$
$$F(Bary(\mathcal{K})) = \{(\sigma_1, \sigma_2, \ldots, \sigma_k) \mid \sigma_i \in F(\mathcal{K}) \ \wedge \ \sigma_1 \subseteq \sigma_2 \subseteq \ldots \subseteq \sigma_k\}.$$

Intuitively, the barycentric subdivision adds a vertex in the center of each simplex, and joins each of the original vertices to the new vertex. Since it does not preserve colors, the more complex chromatic subdivision is typically used instead.

▶ **Definition 37** (Chromatic subdivision)**.** *Let $\mathcal{K}$ be a k-dimensional abstract simplicial complex with a proper coloring $\chi : (V(\mathcal{K})) \to \{1, \ldots, k+1\}$. We define the chromatic subdivision through its vertex set and faces as follows.*

$$V(Ch(\mathcal{K})) = \{(i, \sigma) \mid \sigma \in F(\mathcal{K}) \ \wedge \ i \in \chi(\sigma)\},$$
$$F(Ch(\mathcal{K})) = \{(i_1, \sigma_1), (i_2, \sigma_2), \ldots (i_r, \sigma_r) \mid \sigma_j \in F(\mathcal{K}) \wedge \sigma_1 \subseteq \ldots \subseteq \sigma_r \wedge i_m \neq i_s \text{ for all } m \neq s\}.$$

Intuitively, the chromatic subdivision is similar to the barycentric subdivision, but instead of only inserting a single vertex inside a face, it inserts a smaller face of the same dimension. See Figure 10 for illustrations of both the barycentric and the chromatic subdivision.

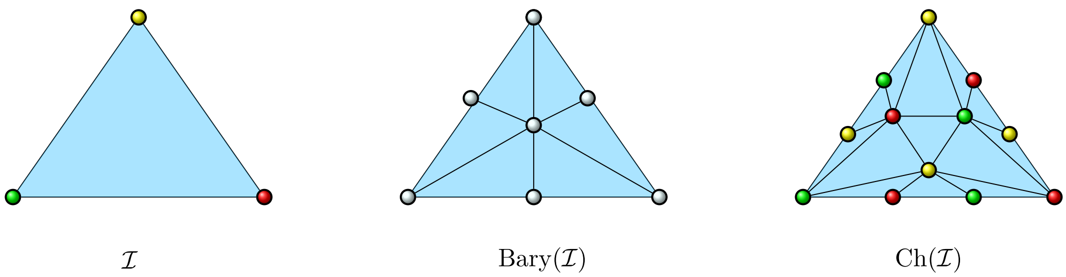


Figure 10 Barycentric vs. chromatic subdivision.

Since we are sometimes interested in relating combinatorial topology with its point-set topology counterpart, we need a precise definition for the topological spaces represented by abstract simplicial complexes. We start with the geometric interpretation of a simplex.

▶ **Definition 38** (Geometric simplex). *Let $\mathcal{K}$ be an abstract simplicial complex, and $\sigma \in F(\mathcal{K})$.*

We define the geometric realization of σ, denoted by $|\sigma|$ as the set of affine functions from σ into the closed interval $[0,1]$. That is $|\sigma| = \{f : \sigma \to [0,1] \mid \sum_{v\in\sigma} f(v) = 1\}$. Elements of $|\sigma|$ are called points of $|\sigma|$.

In order to make proofs more readable, we will also express the points of a geometric simplex in algebraic form, namely, as $\sum_{v\in\sigma} f(v) \cdot v$.

Intuitively, the affine functions form a coordinate system, where the vertices in σ are the generators. A k-dimensional geometric simplex is homeomorphic to a k-dimensional ball, and can be embedded into a k-dimensional Euclidean space. Notice that the above definition has the advantage ob being independent from embeddings, however.

While we have defined the geometric realization of an individual simplex, we also need a point-set topological space for the whole abstract simplicial complex.

▶ **Definition 39** (Geometric Simplicial Complex). *For an abstract simplicial complex $\mathcal{K}$, we define the geometric realization of $\mathcal{K}$, denoted by $|\mathcal{K}|$, as the quotient space $\coprod_{\sigma\in F(\mathcal{K})} |\sigma|/R$. Herein, $\coprod_{\sigma\in F(\mathcal{K})} |\sigma|$ is the disjoint union of all geometric simplices of $\mathcal{K}$, and R is the equivalence relation given by $f \underset{R}{\sim} g$ iff $f^{-1}((0,1]) = g^{-1}((0,1])$ and $f(v) = g(v)$ for all $v \in f^{-1}((0,1])$.*

Via this definition, we create disjoint geometric simplices and glue them along shared points. Observe that the equivalence relation R defines shared points as functions that have the same non-zero coordinates.

We note that a geometric simplicial complex $|\mathcal{K}|$ is homeomorphic to both its barycentric subdivision $|\mathrm{Bary}(\mathcal{K})|$ and to its chromatic subdivision $|\mathrm{Ch}(\mathcal{K})|$. Therefore, it is also homeomorphic to any finitely iterated subdivision.

▶ **Definition 40** (Geometric Star of a Vertex). *Let $v \in V(\mathcal{K})$ be a vertex of an abstract simplicial complex $\mathcal{K}$. We define the closed star of v, $\mathrm{star}(v)$ as the closed subset of $|\mathcal{K}|$ induced by the geometric realization of the simplices that contain v. More precisely, $\mathrm{star}(v) = \bigcup_{v\in\sigma} |\sigma|$. The open star of a vertex is the interior of the closed star, and denoted by $\mathrm{star}^o(v) = \mathrm{int}(\mathrm{star}(v))$; note that $\mathrm{int}(v) = v$ for every vertex. Unless explicitly stated, we will refer to the closed star of a vertex v when using $\mathrm{star}(v)$.*

A.2 The topological model in distributed computing

In the previous subsection, we provided a reasonably comprehensive list of basic definitions of combinatorial topology. In this section, we will use and appropriately extend them in order to be able to model the wait-free asynchronous shared memory (ASM) model. We will focus on the *Iterated Immediate Snapshot* (IIS) Model described in [22], since it is conveniently equivalent to other wait-free shared memory models in terms of computability, and it relates directly to the topological framework.

First, we need to model input and output configurations for distributed computations. This will be done by dedicated simplicial complexes. In the input complex, local input states are modeled as vertices, and global initial configurations as faces. Since we are interested in colored tasks, processes are labeled by their unique ids, which is modeled by a proper coloring of the vertices.

▶ **Definition 41** (Input complex). *We define the input complex of a distributed system, denoted by $\mathcal{I}$, as the following abstract simplicial complex:*

$$V(\mathcal{I}) = \{(p_i, v_i) \mid p_i \text{ is a process id, and } v_i \text{ is a valid input value for } p_i\},$$
$$F(\mathcal{I}) = \{(w_1, \dots, w_n) \mid \text{there is a valid global input configuration that is consistent with all } w_i\}.$$

We can use a symmetrical definition for an output complex.

▶ **Definition 42** (Output complex). *We define the output complex of a distributed system, denoted by $\mathcal{O}$, as the following abstract simplicial complex:*

$$V(\mathcal{O}) = \{(p_i, v_i) \mid p_i \text{ is a process id, and } v_i \text{ is a valid output value for } p_i\},$$
$$F(\mathcal{O}) = \{(w_1, \ldots, w_n) \mid \text{ there is a valid global output configuration that is consistent with all } w_i\}.$$

Carrier maps are used to specify the validity conditions of a task. A simplex in the input complex $\mathcal{I}$, which corresponds to a valid initial configuration, is mapped to a subcomplex of the output complex $\mathcal{O}$. This subcomplex corresponds to the valid different outputs that are allowed for this input.

▶ **Definition 43** (Carrier Set). *For a point $x \in |\mathcal{K}|$, let the carrier of x, denoted $\sigma = carr(x, \mathcal{K})$, be the unique smallest simplex $\sigma \in \mathcal{K}$ such that $x \in |\sigma|$. We can also define the carrier of a set $S \subseteq |\mathcal{K}|$, as $carr(S, \mathcal{K}) = \bigcup_{x \in S} carr(x, \mathcal{K})$.*

Notice that any chromatic subdivision $\mathrm{Sub}(\mathcal{I})$ of a chromatic simplicial complex $\mathcal{I}$ induces a carrier map $\Phi : \mathcal{I} \to \mathrm{Sub}(\mathcal{I})$. Φ maps each simplex σ into a simplicial complex $\mathrm{Sub}(\sigma)$.

The following definition gives the standard meaning of the carrier $\mathrm{carr}(\tau, \Phi)$ of a simplex $\tau \in \mathcal{O}$ for a strict carrier map $\Phi : \mathcal{I} \to 2^{\mathcal{O}}$:

▶ **Definition 44.** *Let $\mathcal{I}, \mathcal{O}$ be finite simplicial complexes and $\Phi : \mathcal{I} \to 2^{\mathcal{O}}$ be a strict carrier map. For each simplex $\tau \in \Phi(\mathcal{I})$, there exists a unique simplex $\sigma \in \mathcal{I}$ of smallest dimension, such that $\tau \in \Phi(\sigma)$. We say that this σ is the carrier of τ under Φ, denoted $carr(\tau, \Phi)$.*

A carrier map $\Delta : \mathcal{I} \to 2^{F(\mathcal{O})}$ can be used to specify the validity conditions for a distributed task, i.e., which outputs are valid for each particular input.

▶ **Definition 45** (Task). *A distributed task T is a triple $\langle \mathcal{I}, \mathcal{O}, \Delta \rangle$, where $\mathcal{I}$ is a pure abstract simplicial complex, $\mathcal{O}$ is a pure abstract simplicial complex of the same dimension as $\mathcal{I}$, and Δ is a carrier map.*

We have now developed a topological model that captures the input and output conditions for a distributed system. However, we are also interested in the evolution of the system throughout an execution. This is modeled by another simplicial complex, the protocol complex.

▶ **Definition 46** (Protocol complex). *We define the protocol complex of a distributed system, denoted by $\mathcal{P}$, as the following abstract simplicial complex:*

$$V(\mathcal{P}) = \{(p_i, s) \mid p_i \text{ is a process id, and } s \text{ is a valid local state for } p_i\},$$
$$F(\mathcal{P}) = \{(w_1, \ldots, w_n) \mid \text{ there is a valid global state configuration that is consistent with all } w_i\}.$$

In contrast to the input and output complexes, which are very well defined, there is no standard way of describing the protocol complex, as it intrinsically depends on the peculiarities of the computing model, like message passing versus shared memory communication, asynchronous versus synchronous execution etc. In fact, it is the communication between processes over time that determines the evolution of the protocol complex, which starts out from the input complex.

In the IIS model, all protocol complexes are chromatic subdivisions of $\mathcal{I}$. More specifically, an execution with k communication rounds is represented by the the k-th chromatic subdivision of $\mathcal{I}$, that is, $\mathcal{P} = \mathrm{Ch}^k(\mathcal{I})$.

We can now finally state what it means to solve a task in the topological model.

▶ **Definition 47** (Task solvability)**.** *We say that a task* $T = \langle \mathcal{I}, \mathcal{O}, \Delta \rangle$ *is solvable in a distributed model* $\mathcal{M}$*, if there exists a protocol complex* $\mathcal{P}$ *associated to the model* $\mathcal{M}$ *and a chromatic simplicial map* $\mu : \mathcal{P} \to \mathcal{O}$ *that is carried by* Δ*.*