mathematics

MDPI

Article

Asynchronous Computability Theorem in Arbitrary Solo Models

Yunguang Yue [1,†], Fengchun Lei [1,*,†], Xingwu Liu [1,2,3,†] and Jie Wu [4,†]

1 School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China; yueyunguang1988@mail.dlut.edu.cn (Y.Y.); liuxingwu@ict.ac.cn (X.L.)
2 SKL Computer Architecture, ICT, CAS, Beijing 100049, China
3 School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China
4 College of Mathematical Sciences, Hebei Normal University, Shijiazhuang 050024, China; wujie@hebtu.edu.cn
* Correspondence: fclei@dlut.edu.cn
† These authors contributed equally to this work.

Received: 2 March 2020; Accepted: 29 April 2020; Published: 10 May 2020

check for updates

Abstract: In this paper, we establish the asynchronous computability theorem in d-solo system by borrowing concepts from combinatorial topology, in which we state a necessary and sufficient conditions for a task to be wait-free computable in that system. Intuitively, a d-solo system allows as many d processes to access it as if each were running solo, namely, without detecting communication from any peer. As an application, we completely characterize the solvability of the input-less tasks in such systems. This characterization also leads to a hardness classification of these tasks according to whether their output complexes hold a d-nest structure. As a byproduct, we find an alternative way to distinguish the computational power of d-solo objects for different d.

Keywords: distributed computing; asynchronous computability; solo model; solvability; combinatorial topology

1. Introduction

Asynchronous computability, which means the solvability of tasks in asynchronous, failure-prone distributed systems, has been an active topic ever since 1985 when the well known FLP impossibility theorem was established [1]. There is a long line of work that deals with this topic for numerous tasks in different systems under various failure models [2–10]. However, only a pinch of results exist that characterize the asynchronous computability of general, rather than specific, tasks.

Such efforts date back to 1988, when Biran et al. [11] established a graph-theoretical sufficient and necessary condition for the solvability of distributed tasks in message-passing systems. Their method applies only if there is at most one crash failure of the process. Breakthrough was made in 1993 by three independent teams [5,6,12], among which Herlihy and Shavit [6] presented a topological framework for studying the asynchronous computability of general tasks in share-memory or message-passing systems with $t \geq 1$ crash failures. This framework was then extended to a complete characterization of wait-free solvability of distributed tasks in shared-memory systems, namely, a task is solvable if and only if its specification is topologically compatible in some sense [7]. The characterization was further generalized in two directions. One direction is generalization to systems with arbitrary communication objects (not just shared-memory of message passing) [1,13], to arbitrary resilience (rather than one or n failures) [6,8,14], to arbitrary synchrony [15,16], or to Byzantine failures that may do the most malicious harm on the system [9,17,18]. The other direction is to characterize the relative hardness of distributed tasks in asynchronous shared-memory systems: two tasks are reducible to each other if and only if they are equivalent in a topological manner [19–22].

Closely related to computability is the concept of decidability, which means whether there is an efficient procedure to decide the solvability of a family of tasks in a system model. It was initiated by Herlihy and Rajsbaum in [19] and continued in [20,23], where shared-memory and communication objects of consensus number more than one are considered.

The above-mentioned works mainly assume simple communication objects such as shared-memory and message passing. The two objects were generalized by Herlihy et al. in 2014 [24] into a spectrum of d-solo objects, where d is a positive integer ranging over the set $\{1, \ldots, n+1\}$ and n is the number of processes. Roughly speaking, a d-solo communication object allows up to d processes to run *solo*, that is none of them gets information from any peer process. The d-solo execution comes from a real-life distributed computing decision scenarios, the robot gathering problem [25–28]. A robot gathering problem of a set of n asynchronous robots requires that robots can move to meet in a given area of some space (e.g., [29,30]) which depends on their initial positions. Since the initial position of a robot constitutes its input value and crash-prone robots may appear asynchronously in the space in the Look–Compute–Move model which corresponds to a wait-free snapshot shared-memory model [31] where robots can take snapshots of the graph where they are located [32], and a robot cannot view the positions of all other robots instantaneously in reality, for one robot looks in one direction, it could be that the other robot just moved out of the corresponding region, then it is possible that two robots don't see each other during the execution of their Look–Compute–Move, which is a situation similar to d-solo executions.

It is not hard to see that shared-memory and message passing lie at the extremes of the spectrum with $d = 1$ and $d = n + 1$, respectively. Then a question naturally arises: is it possible to generalize the asynchronous computability theory from the extremes to the whole spectrum?

To date, little has been known about the answer, except for the progress made by Herlihy et al. [10]. However, Herlihy et al. [10] only investigated the solvability of colorless tasks, which belong to a rather constrained class of distributed tasks introduced by Borowsky et al. in [33]. The mission of our paper is to extend their work to general tasks, so as to completely answer the question.

Our contribution lies in three aspects. First, we fully characterize the asynchronous computability of distributed decision tasks on d-solo systems for arbitrary $1 \leq d \leq n+1$. The characterization is in terms of a topological property of the task's specification, strengthening the bridge between topology and computing. Second, as an application, we derive a simple necessary and sufficient condition for the solvability of input-less tasks in d-solo systems. input-less tasks were proposed by Gafni et al. in [23]. They are among the simplest distributed tasks, and play a critical role in research on distributed computability and decidability. Third, we identify a hierarchy of input-less tasks, which exactly differentiates the computational power of d-solo objects for different d. This provides an alternative to the more sophisticated agreement-like tasks constructed by Herlihy et al. [10].

2. Asynchronous Computability Theorem in d-Solo Models

2.1. Computational Model of Distributed Computing

In this subsection, we review briefly review some concepts of distributed computing. For details, please refer to [6,7,13,15,19,20,23,34].

A distributed system consists of $n + 1$ sequential processes $\Pi = \{p_0, p_1, \ldots, p_n\}$ and some communication objects through which the processes communicate. Without loss of generality, assume that each process proceeds round by round. We further assume that the system is asynchronous and that up to n processes may fail by crashing. In any execution, each process starts with a private input value, and if non-faulty, ends with a private output value after a finite sequence of communications and local computations. A task is a specification of eligible outputs with regard to the inputs, which intuitively models a coordination problem.

Now, we recall the d-solo communication object $CO^d[r]$ proposed by Herlihy et al. [24], where integers $1 \leq d \leq n+1$ and $r \geq 1$ stand for the solo-dimension and the round number, respectively. Roughly speaking, $CO^d[r]$ only allows processes in round r to communicate, and up to d of the processes can run in solo, i.e., each gets no information from its peers. The behavior of $CO^d[r]$ is depicted as follows. Let $(\Pi_0, \ldots, \Pi_z)$ be an ordered pairwise-disjoint partition of a non-empty subset $\Pi' \subseteq \Pi$, such that $|\Pi_0| \leq d$ and $\Pi_i \neq \emptyset$ for any $1 \leq i \leq z$. It represents the following communication result: each process $p_j \in \Pi'$ reaches round r and invokes $CO^d[r]$ on some value v_j, and the invocation returns v_j to p_j if $p_j \in \Pi_0$, while it returns $\{v_l | p_l \in \bigcup_{0 \leq k \leq i} \Pi_k\}$ if $p_j \in \Pi_i$ with $i > 0$. Intuitively, Π_0 is the set of processes that run in solo in round r. For more detail, refer to the literature [10].

A protocol is a distributed program consisting of the processes. A protocol is said to solve a task if the outputs of each execution conform with the specification of the task. Full-information protocols are widely studied, where each process keeps communicating without local computation until it decides. Full-information protocols play a key role in distributed computing, because they are universal in the sense that a task is solvable if and only if it can be solved by a full-information protocol. When processes communicate via d-solo object CO^d, a full-information protocol has the form illustrated in Table 1. Here, $decidable_i(\cdot)$ is a boolean function indicating whether or not the process p_i can make a decision based on the local state ls_i, and $\delta_i(\cdot)$ returns p_i's decision value. What we need to know is that p_i makes a decision and $\delta_i(\cdot)$ returns just once if it can decide.

Table 1. Process p_i for a full-information protocol.

$r_i \leftarrow 0$, $ls_i \leftarrow$ initial local state;
loop forever
    $r_i \leftarrow r_i + 1$;
    $view_i \leftarrow CO^d[r_i].communicate(i, ls_i)$;
    $ls_i \leftarrow (ls_i, view_i)$;
    if $decidable_i(ls_i)$ **then** output $\delta_i(ls_i)$ **end if**
end loop

2.2. *Topology Model of Distributed Computing*

In this subsection, we review concepts of combinatorial topology briefly, and then we model distributed computing systems in terms of combinatorial topology. One can refer to [34–37] for further details.

2.2.1. Simplicial Complex and Simplicial Map

An *abstract simplicial complex* $\mathcal{K}$, or complex for short, is a collection of non-empty subsets of a finite set. The complex is required to be closed under containment, meaning that if $\alpha \in \mathcal{K}$, then all non-empty subsets of α are included in $\mathcal{K}$. A member of $\mathcal{K}$ is called a simplex. Given two simplexes $\alpha, \beta \in \mathcal{K}$, we say that α is a $face$ of β if $\alpha \subseteq \beta$, and we say that α is a proper $face$ of β if $\alpha \subset \beta$. Define the dimension of a simplex α to be $dim(\alpha) \triangleq ||\alpha|| - 1$, where $||\alpha||$ stands for the cardinal number of the set α. The dimension of a simplicial complex is the highest dimension among its simplexes. We simply call an n-dimensional simplex α an n-simplex, and use the superscript form α^n to indicate the dimension when needed. The notation is likewise defined on complexes. Any 0-simplex of a complex $\mathcal{K}$ is called a vertex of $\mathcal{K}$, and we use $V(\mathcal{K})$ to denote the set of vertices of $\mathcal{K}$. A n-complex $\mathcal{K}$ is said to be pure, if any simplex $\alpha \in \mathcal{K}$ is a face of some n-simplex $\beta \in \mathcal{K}$. By default, we only consider pure complexes in this paper. Any complex $\mathcal{K}' \subseteq \mathcal{K}$ is called a subcomplex of $\mathcal{K}$. Given a pure n-complex $\mathcal{K}$, any $(n-1)$-simplex α which is a face of a unique n-simplex is called a boundary simplex, and the subcomplex consisting of all boundary simplexes and their faces is called the boundary of $\mathcal{K}$, denoted by $\partial(\mathcal{K})$.

To ease understanding, one can equivalently view abstract simplicial complex $\mathcal{K}$ through the geometric lens which we call geometric simplicial complex. Bijectively map $V(\mathcal{K})$ to an arbitrary set of affinely independent points in an Euclidean space. Each simplex naturally corresponds to the convex hull spanned by the images of its vertices. Seeing Figure 1, a vertex is 0-simplex, a line segment is a 1-simplex, a triangle is a 2-simplex and a tetrahedron is a 3-simplex. Putting the convex hulls together, we get a geometric realization $|\mathcal{K}|$ of $\mathcal{K}$. Nevertheless, hereunder, we still adopt the definition of the abstract simplicial complex.

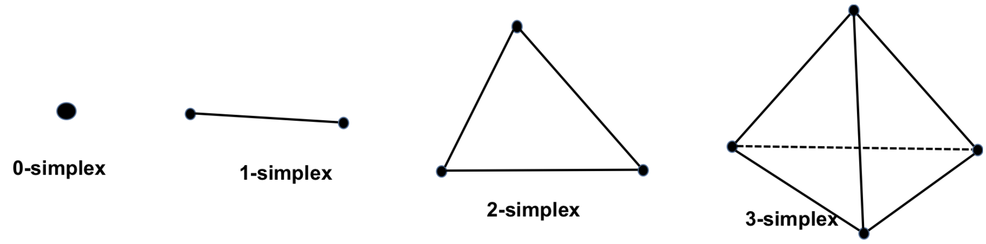

Figure 1. Low dimensional simplexes.

Given complexes $\mathcal{K}$ and $\mathcal{K}'$, a map $\mu : V(\mathcal{K}) \longrightarrow V(\mathcal{K}')$ is said to be a simplicial map from $\mathcal{K}$ to $\mathcal{K}'$ if it map every simplex in $\mathcal{K}$ to a simplex in $\mathcal{K}'$. A map $\Psi : \mathcal{K} \longrightarrow 2^{\mathcal{K}'}$ is called a carrier map from $\mathcal{K}$ to $\mathcal{K}'$. Given a simplicial map μ and a carrier map from $\mathcal{K}$ to $\mathcal{K}'$, we say that μ is carried by Ψ if $\psi(\alpha) \in \Psi(\alpha)$ for any simplex $\alpha \in \mathcal{K}$.

Let $\mathcal{K}$ and $\mathcal{K}'$ be two complexes, we say that $\mathcal{K}$ is isomorphic to $\mathcal{K}'$ if there exist simplicial maps $f : \mathcal{K} \longrightarrow \mathcal{K}'$ and $g : \mathcal{K}' \longrightarrow \mathcal{K}$ such that the compositions $g \circ f$ and $f \circ g$ are both identity maps.

Let $\mathcal{K}$ be a simplicial complexes and $\mathcal{C}$ be a finite set. We say $\mathcal{K}$ is $\mathcal{C}$-chromatic if there is a map $\chi : V(\mathcal{K}) \longrightarrow \mathcal{C}$ such that $\chi(u) \neq \chi(v)$ for any vertices u, v of a common simplex. We call χ a $\mathcal{C}$-coloring of $\mathcal{K}$ and use $\chi(\alpha)$ to denote the set of colors of the vertices of a simplex α. Suppose complexes $\mathcal{K}$ and $\mathcal{K}'$ have $\mathcal{C}$-colorings χ and χ', respectively. A map $\mu : V(\mathcal{K}) \longrightarrow V(\mathcal{K}')$ is called color-preserving if $\chi(v) = \chi'(\mu(v))$ for any $v \in V(\mathcal{K})$.

For any integer $n \geq 0$, we will use $[n]$ to stand for the set $\{0, 1, 2, \ldots, n\}$.

Let α be an n-simplex and χ be a $\mathcal{C}$-coloring of α. The *standard chromatic subdivision* of α, denoted by $Ch(\alpha)$, is the n-complex $Ch(\alpha)$ constructed as follows:

1. $V(Ch(\alpha)) = \{(c, S) : S \text{ is a face of } \alpha, c \in \chi(S)\}$;
2. Any set of vertices $\{(c_0, S_0), (c_1, S_1), \ldots, (c_m, S_m)\}$ is a simplex of $Ch(\alpha)$ if and only if both the following conditions are satisfied for any $i \neq j \in [m]$:
    - $c_i \neq c_j$, and either $S_i \subseteq S_j$ or $S_j \subseteq S_i$;
    - If $c_i \in \chi(S_j)$ then $S_i \subseteq S_j$.

Note that $Ch(\alpha)$ has a natural $\mathcal{C}$-coloring: color any vertex (c, S) in c. Hence $Ch(\alpha)$ is also $\mathcal{C}$-chromatic.

Figure 2 illustrates the barycenter subdivision which has a detailed description in [37], and the standard chromatic subdivision of a 2-simplex, viewing through the geometric lens.

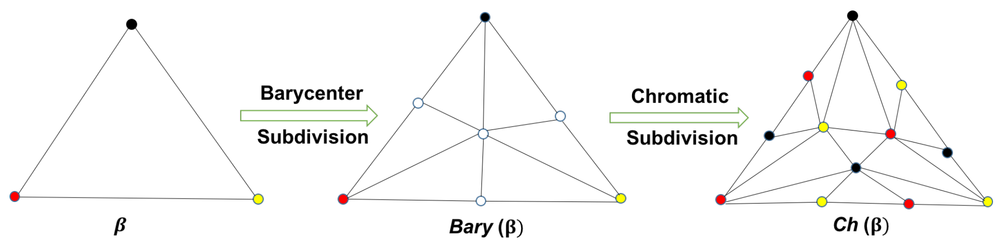

Figure 2. Once barycenter subdivision and once standard chromatic subdivision of a 2-simplex β.

2.2.2. Task Formalization

Formally, a distributed task T on $n+1$ processes can be modeled as a triple $(\mathcal{I}, \mathcal{O}, \Delta)$, where $\mathcal{I}$ and $\mathcal{O}$ are pure n-complexes and Δ is a carrier map from $\mathcal{I}$ to $\mathcal{O}$. Δ is dimension-preserving in the sense that for any $\alpha \in \mathcal{I}$, $\Delta(\alpha)$ is a pure complex of dimension $dim(\alpha)$. The n-simplexes in the input complex $\mathcal{I}$ are such $\{(0, u_0), \ldots, (n, u_n)\}$ where u_i ranges over the input domain of process p_i. Likewise for $\mathcal{O}$, except that the values range over the output domains. With any vertex (i, x) colored by the color i, both $\mathcal{I}$ and $\mathcal{O}$ are $[n]$-chromatic complexes. Δ must satisfies that $\chi(v) = \chi(\Delta(v))$ for any 0-simplex $v \in \mathcal{I}$. Consider m-simplices $\alpha = \{(i_0, u_0), \ldots, (i_m, u_m)\} \in \mathcal{I}$ and $\beta = \{(i_0, v_0), \ldots, (i_m, v_m)\} \in \mathcal{O}$. Then, $\beta \in \Delta(\alpha)$ means when only processes $p_{i_0}, \ldots, p_{i_m}$ participate in an execution, if each p_{i_j} gets input u_j, it is eligible that each p_{i_j} ends with output v_j for $j \in [m]$.

Example 1. *As shown in Figure 3, it is a standard input-less 2-task $T = (\mathcal{I}^2, \mathcal{O}^2, \Delta)$, in which the left part of this Figure represents the input complex $\mathcal{I}^2$ and the right part represents the output complex. The carrier map Δ is illustrated by the dash arrows. Specifically, it maps each vertex A, B, C to a_0, b_0, c_0, each 1-simplex $\{u, v\}$ to $\{\Delta(u), \Delta(v)\}$, and the 2-simplex to $\mathcal{O}$. Intuitively, this means that if some process does not participate, the participating ones have a unique output; otherwise, any outputs characterized by $\mathcal{O}$ are eligible.*

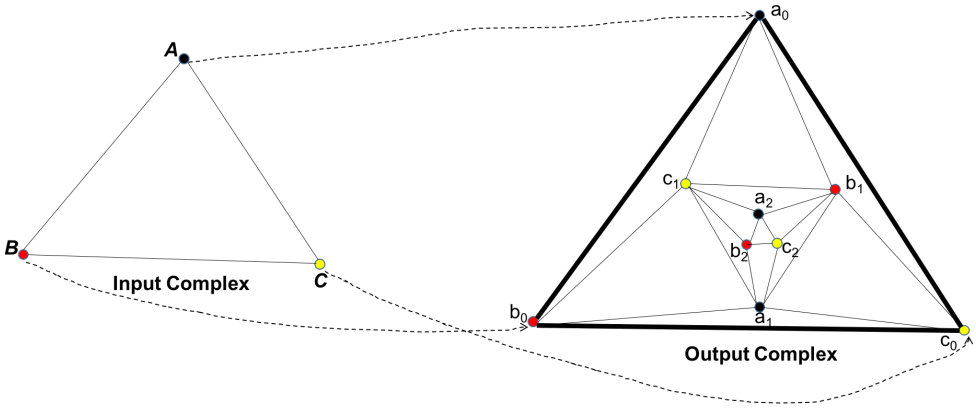

Figure 3. A standard input-less 2-task.

2.2.3. Protocol Complex

Consider a full-information protocol $\mathcal{P}$ as shown in Table 1. Fix an input simplex α. In an execution, if each process p_i finishes round k with local state ls_i, the global state of $\mathcal{P}$ at the end of round k in this execution is modeled as the simplex $\beta = \{(i, ls_i) : 0 \leq i \leq m\}, m \leq n$. We say that β is reachable from α by $\mathcal{P}$. All such simplexes, together with all their faces, constitute the k-round protocol complex of $\mathcal{P}$ reachable from α, denoted by $\mathcal{P}^k(\alpha)$. Let k be the last round before making decision. Then complex $\mathcal{P}(\mathcal{I}) \triangleq \bigcup_{\alpha \in \mathcal{I}} \mathcal{P}^k(\alpha)$ is called the protocol complex of $\mathcal{P}$.

Example 2. *Figure 4 illustrates the geometric view of the 1-round protocol complex of the full-information protocol on three processes. The dots are colored yellow, black and green to stand for the processes p_0, p_1 and p_2. Consider the communication objects CO^d with $d = 1, 2, 3$, respectively. An arrow represents that the destination process has received the value communicated via CO^d by the source process. Each filled triangle represents a 2-simplex, modeling a reachable global state. The collection of all the 2-simplexes in the left part of the Figure 4 represents 1-round protocol complex when the communication object is 1-solo. The middle part of Figure 4 shows the extra 2-simplexes that appear when the communication object changes from 1-solo to 2-solo. Hence, the simplexes in the middle and those in the left part of this figure form the 1-round protocol complex when $d = 2$ in CO^d. If we further increase d to 3, one more 2-simplex appears, as illustrated in the right part of Figure 4, which represents an execution in which all of the three processes run in solo. Putting all the simplexes together, we get the 1-round protocol complex when CO^3 is used.*

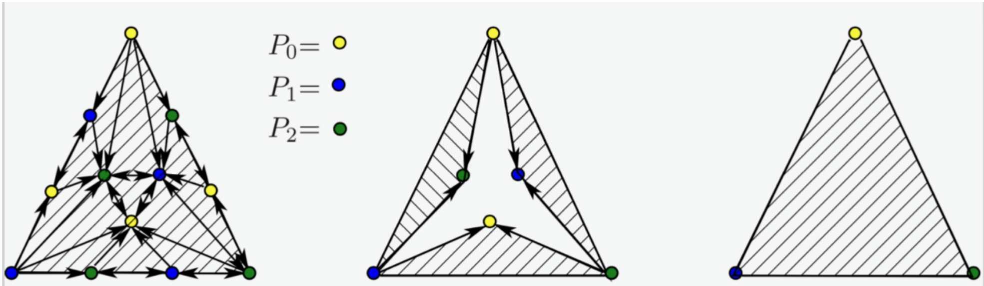

Figure 4. All possible execution (filled 2-simplexes) for three processes.

2.3. Chromatic Join

In this subsection, we mainly define a chromatic complex, a chromatic join, which can describe the execution of d-solo model accurately. Then we introduce a simple property of the complex.

Given two sets X_0 and X_1 with no intersection, call $X_0 * X_1$ the *join* of X_0 and X_1 if $X_0 * X_1 \triangleq X_0 \sqcup X_1$.

Let α be a chromatic simplex. Consider a non-empty set $S \subseteq V(Ch(\alpha))$. Define the low-dimensional part $L(S) \triangleq \{(c, T) \in S : dim(T) = 0\}$, and the high-dimensional part $H(S) \triangleq S \setminus L(S)$. We say that S is compatible if both the following conditions hold:

- For any $(c, T) \neq (c', T') \in S$, $c \neq c'$;
- $\{u\} * H(S)$ is a simplex in $Ch(\alpha)$ for any $u \in L(S)$.

Roughly speaking, S, if compatible, stands for the global state resulted from a 1-round execution of the full-information protocol, where processes in $L(S)$ run in solo.

Definition 1. *Given a chromatic simplex α and an integer $d \geq 1$, the chromatic d-join of α is defined to be*

$$\mathbb{D}_d(\alpha) \triangleq \{S \subseteq V(Ch(\alpha)) : S \text{ is compatible}, ||L(S)|| \leq d\}.$$

Remark 1. *We can regard S as a join $L(S) * H(S)$ of $L(S)$ and $H(S)$ that are subsets of $V(Ch(\alpha))$ for any element $S \in \mathbb{D}_d(\alpha)$;*

In the definition of chromatic d-join, S can be such that $||L(S)|| \leq 1$. This implies that $Ch(\alpha) \subseteq \mathbb{D}_d(\alpha)$.

Now we extend the definition to complexes. Suppose $\mathcal{K}$ is a chromatic n-complex. Its chromatic d-join is defined to be $\mathbb{D}_d(\mathcal{K}) \triangleq \bigcup_{\alpha \in \mathcal{K}} \mathbb{D}_d(\alpha)$.

Lemma 1. *$\mathbb{D}_d(\mathcal{K})$ is a chromatic simplicial complex.*

Proof of Lemma 1. We first show that $\mathbb{D}_d(\mathcal{K})$ is closed under containment and intersection.

Let β be any simplex in $\mathbb{D}_d(\mathcal{K})$, then there is a simplex $\alpha \in \mathcal{K}$ such that $\beta = L(S) * H(S)$ in which S is in $\mathbb{D}_d(\alpha)$. Suppose β' is any face of β. There are three cases: $V(\beta') \subseteq L(S)$ or $V(\beta') \subseteq H(S)$, or $V(\beta') \cap H(L) \neq \emptyset$ and $V(\beta') \cap H(S) \neq \emptyset$. It is not hard to show that in each case of them, we can regard β' as a d-join of a subset of $L(S)$ and a subset of $H(S)$. It follows that β' is in $\mathbb{D}_d(\alpha)$, and it is also in $\mathbb{D}_d(\mathcal{K})$.

Let γ be any another simplex in $\mathbb{D}_d(\mathcal{K})$. If the intersections of γ and β is not empty set, then it must be a common face of them. By the upper argument, we know that it is also in $\mathbb{D}_d(\mathcal{K})$.

Then label each vertex $(c, T) \in V(\mathbb{D}_d(\mathcal{K}))$ with c. From the definition of $\mathbb{D}_d(\mathcal{K})$, it is obvious that this labeling is a coloring map of $\mathbb{D}_d(\mathcal{K})$. □

Throughout this paper, whenever coloring of $\mathbb{D}_d(\mathcal{K})$ is mentioned, it always means the labeling in the proof.

Inductively, we define the s-fold chromatic d-join $\mathbb{D}_d^s(\mathcal{K}) \triangleq \mathbb{D}_d(\mathbb{D}_d^{s-1}(\mathcal{K}))$, for any positive integer s. Particularly, if $s = 0$, we set $\mathbb{D}_d^0(\mathcal{K}) \triangleq \mathcal{K}$.

From Lemma 1, we note that $\mathbb{D}_d^s(\alpha)$ is chromatic simplicial complex for any given chromatic simplex α, then we can regard $\mathbb{D}_d^s$ as a carrier map. On this ground, we define the *carrier* of a simplex in $\mathbb{D}_d^s(\alpha)$.

Definition 2. *Given a chromatic simplicial complex $\mathcal{K}$, consider an arbitrary simplex that $\beta \in \mathbb{D}_d(\mathcal{K})$. Let τ be the minimal simplex in $\mathcal{K}$ such that $\beta \in \mathbb{D}_d(\tau)$. Then τ is called β's carrier in $\mathcal{K}$ under carrier map $\mathbb{D}_d$, denoted by $carrier(\beta, \mathcal{K})$. Inductively, we can define the carrier of $\beta \in \mathbb{D}_d^s(\mathcal{K})$, that is, $carrier^s(\beta, \mathcal{K}) = carrier^{s-1}(carrier(\beta, \mathbb{D}^{s-1}(\mathcal{K})), \mathcal{K})$.*

Suppose $\beta = L(S) * H(S)$, then $carrier(\beta, \mathcal{K}) = L(S)$ if $H(S) = \emptyset$, otherwise, there must be a vertex $(c, T) \in H(S)$ such that for any vertex $(c', T') \in H(S)$, $T' \subseteq T$, then $carrier(\beta, \mathcal{K}) = T$.

The next lemma indicates that the d-join $\mathbb{D}_d^s(\cdot)$ preserves some connectedness property.

Lemma 2. *Let $\mathcal{K}$, $\mathcal{K}'$ be two $[n]$-chromatic n-complexes. If there is a color-preserving simplicial map $f : \mathbb{D}_d^k(\mathcal{K}) \longrightarrow \mathcal{K}'$ for non-negative integers k and d, there is a color-preserving simplicial map $g : \mathbb{D}_d^{k+1}(\mathcal{K}) \longrightarrow \mathcal{K}'$.*

Proof of Lemma 2. Let $\alpha = \{(i, v_i)\}_{i\in[n]}$ be any n-simplex of $\mathbb{D}_d^k(\mathcal{K})$. Since f is color-preserving simplicial map, then $f(\alpha)$ is a chromatic n-simplex of $\mathcal{K}'$ with the same colors of α. Let g_α be a point-to-point map from $V(\mathbb{D}_d(\alpha))$ to $V(f(\alpha))$ such that

$$g_\alpha((i, T_{r_i})) = f((i, v_i))$$

for any $(i, T_{r_i}) \in V(\mathbb{D}_d(\alpha))$, $i, r_i \in [n]$. We only need to check that g_α is simplicial map from $\mathbb{D}_d(\alpha)$ to $\mathcal{K}'$.

Let β is any n-simplex of $\mathbb{D}_d(\alpha)$, since $\mathbb{D}_d(\alpha)$ is chromatic complex by Lemma 1, then we can assume that $\beta = \{(i, T_{r_i})\}_{i\in[n]}$. It follows that

$$\begin{aligned} g_\alpha(\beta) &= g_\alpha(\{(i, T_{r_i})\}_{i\in[n]}) \\ &= \{g_\alpha((i, T_{r_i}))\}_{i\in[n]} \\ &= \{f((i, v_i))\}_{i\in[n]} \\ &= f(\{(i, v_i)\}_{i\in[n]}) \\ &= f(\alpha) \end{aligned}$$

which is an n-simplex in $\mathcal{K}'$. Since β is chosen arbitrarily, then g_α is a color-preserving simplicial map from $\mathbb{D}_d(\alpha)$ to $\mathcal{K}'$.

Assume α' is another n-simplex of $\mathbb{D}_d^k(\mathcal{K})$ and τ is the common face of α and α', then

$$\begin{aligned} \mathbb{D}_d(\alpha) \cap \mathbb{D}_d(\alpha') &= \mathbb{D}_d(\alpha \cap \alpha') \\ &= \mathbb{D}_d(\tau) \end{aligned}$$

It follows that $g_\alpha(\mathbb{D}_d(\tau))$ is equal to $g_{\alpha'}(\mathbb{D}_d(\tau))$ by former argument. As a result, all such maps agree on their intersections. We just constructed

$$g = \sqcup_{\alpha\in\mathbb{D}_d^k(\mathcal{K})} g_\alpha$$

then g is a color-preserving simplicial map such that for any simplex β of $\mathbb{D}_d^{k+1}(\mathcal{K})$, $g(\beta)$ is equal to $g_{\beta'}(\beta)$, where β' is $carrier(\beta, \mathbb{D}_d^k(\mathcal{K}))$. Note that α is arbitrary and $\mathbb{D}_d^{k+1}(\mathcal{K})$ is $\mathbb{D}_d(\mathbb{D}_d^k(\mathcal{K}))$, it follows that g is a color-preserving simplicial map from $\mathbb{D}_d^{k+1}(\mathcal{K})$ to $\mathcal{K}'$. □

Remark 2. *Let us explain the relation of maps g_α and g. Since we have shown that g_α is a simplicial map for any α in $\mathbb{D}_d^k(\mathcal{K})$, then it is a "linear" map [38], which makes that the images of the intersection under g_α and g_β are the same if α has an intersection with β. Then we can regard g as an attaching map of all pieces g_α.*

By this lemma, we know that if there is a color-preserving simplicial map from $\mathbb{D}_d^k(\mathcal{K})$ to $\mathcal{K}'$, then there also exists a color-preserving simplicial map from $\mathbb{D}_d^{k'}(\mathcal{K})$ to $\mathcal{K}'$ for any integer $k' \geq k$.

2.4. Main Theorem

In this subsection, we first show the connection between protocol complex and chromatic d-join of the input complex, and then we prove one of main theorems.

Lemma 3. *When the input complex is $\mathcal{I}$, the protocol complex $\mathcal{P}^1(\mathcal{I})$ of any 1-round full-information protocol in d-solo model is exactly the chromatic d-join $\mathbb{D}_d(\mathcal{I})$.*

Proof of Lemma 3. The basic strategy of the proof is to show that there exists a one-to-one correspondence between the set of protocol complexes on $\mathcal{I}$ of any protocol $\mathcal{P}$ with 1-round and the once chromatic d-join of $\mathcal{I}$ in Definition 1, where $\mathcal{I}$ is the input complex colored by χ with colors $[n]$. Roughly speaking, we firstly show that given an input simplex of $\mathcal{I}$ and an execution of the protocol $\mathcal{P}$ arbitrarily, the output complex is a subcomplex of $\mathbb{D}_d(\mathcal{I})$. Secondly, we need to show that given any simplex O of $\mathbb{D}_d(\mathcal{I})$, there must exist an execution and an input simplex such that there is a possible output which corresponds to O.

Consider any maximal input n-simplex $I = \{x_0, x_1, \ldots, x_n\}$ in $\mathcal{I}$ as an initial global state, where $x_i = (i, v_i)$, $i \in [n]$ in which i and v_i stand for the ID and the input value of a participating process p_i. Let $\Pi = \{p_i\}_{i\in[n]}$ be the set of processes that are coming from I. Consider any execution ϵ of protocol $\mathcal{P}$ with 1-round in d-solo object CO^d. From the behavior of object CO^d, ϵ factly gives a sequence of all processes along the order that run CO^d, denoted by $p_{t_0} \preceq p_{t_1} \preceq \cdots \preceq p_{t_n}$, where $t_j, j \in [n]$. That is if $p_{t_i} \preceq p_{t_j}$ then process p_{t_i} runs object CO^d before p_{t_j} or both of them run concurrently. Define an ordered partition of Π by this sequence, denoted by $\{\pi_0, \pi_1, \ldots, \pi_q\}$, such that all processes in π_j run object CO^d concurrently and processes in π_i run object CO^d before processes in π_j for $0 \leq i < j \leq q$. It follows that processes that executing solo appear only in π_0. Let π_0 be an empty set if there is no process that runs solo. Note that at most d processes run solo, then $0 \leq |\pi_0| \leq d$. Assume that all processes of Π finishing ϵ have returns, denoted by

$$S = \{(i, l_i) | p_i \in \pi_j, 0 \leq j \leq q\}$$

Note that if $p_i \in \pi_0$, process p_i only see itself then $l_i = \{(i, v_i)\}$ which is a 0-dimensional face of I, and if $p_i \in \pi_s$, $0 < s \leq q$, process p_i will see all processes that run before and concurrently, then

$$l_i = \{(j, v_j) | p_j \in \pi_r, r \leq s\}$$

is a non-zero dimensional face of I. By the definition of standard chromatic subdivision, $(i, l_i) \in V(Ch(I))$ for any $(i, l_i) \in S$, then $S \subseteq Ch(I)$.

Let $L(S)$ be equal to $\{(i, l_i) \in S : dim(l_i) = 0\}$ and let $H(S)$ be complementary set of $L(S)$ in S, then for any $(i, l_i) \in L(S)$ and any $(j, l_j), (r, l_r) \in H(S)$, we have

$$l_i \subset l_j \subseteq l_r \text{ or } l_i \subset l_r \subseteq l_j$$

It follows that S is compatible. Note that $||L(S)|| \leq d$, then S is a d-join and in $\mathbb{D}_d(I)$ by Definition 1. If not all processes have returns, that is, some processes may be faulty, then the returns $\widetilde{S}$ is a subset of S, for all processes participate and deposit their initial local state in object CO^d. Then $\widetilde{S}$ is also a d-join and in $\mathbb{D}_d(I)$. Since I and ϵ are arbitrary, then $\mathcal{P}^1(\mathcal{I}) \subseteq \mathbb{D}_d(\mathcal{I})$.

On the contrary, given any m-simplex $\alpha' = \{(j, T_j)\}_{j \in A'}$ of $\mathbb{D}_d(\mathcal{I})$. Suppose $\alpha = carrier(\alpha', \mathcal{I})$, then T_j is a face of α for any $j \in A'$. Let $A = \chi(\alpha)$, then $A' \subseteq A$. Let $\Pi = \{p_i | i \in A\}$ and let $\Pi' = \{p_i | i \in A'\}$, then $\Pi' \subseteq \Pi$.

Since α' is a d-join, then it is compatible and it has form $L(\alpha') * H(\alpha')$ with $||L(\alpha')|| \leq d$, where $L(\alpha')$ and $H(\alpha')$ are the low-dimensional part and the high-dimensional part of α'. And then for any vertex $u \in L(\alpha')$, $\{u\} \sqcup H(\alpha')$ is a face of $Ch(\alpha)$. It follows that we can give a partition of the set of classes over all processes of Π', denoted $\{\pi_0, \pi_1, \ldots, \pi_s\}$, such that, if $(i, T_i) \in L(\alpha')$ then we put p_i into π_0; and for any two processes $(c, T_c), (j, T_j) \in H(\alpha')$, if $T_c = T_j$ then we put them in a same class π_t with $0 < t \leq s$, and if $T_c \subset T_j$ we put p_c and p_j into π_{s_c} and π_{s_j} such that $0 < s_c < s_j \leq s$.

Next, we will give an ordered partition of Π by Π'. Step 1. For any one $p_i \in \pi_1$, check (i, T_i), if

$$dim(T_i) = ||\pi_0|| + ||\pi_1|| - 1$$

do nothing; if

$$dim(T_i) > ||\pi_0|| + ||\pi_1|| - 1$$

that is, there exists at least one process that is in $\{p_x | x \in \chi(T_i)\}$ but not in Π', then put

$$\{p_x | x \in \chi(T_i)\} - (\pi_0 \cup \pi_1)$$

together as a new class π_{0_0} between π_0 and π_1. Step 2. For any one $p_j \in \pi_t$, $2 \leq t \leq s$, check (j, T_j), if

$$dim(T_j) = \sum_{i=0}^{t-1}(||\pi_i|| + ||\pi_{i_0}||) + ||\pi_t|| - 1$$

then do nothing; if

$$dim(T_j) > \sum_{i=0}^{t-1}(||\pi_i|| + ||\pi_{i_0}||) + ||\pi_t|| - 1$$

that is, there exists at least one process that is in $\{p_x | x \in \chi(T_j)\}$ but not in Π', then put

$$\{\{p_x | x \in \chi(T_i)\} - \cup_{x=0}^{t-1}(\pi_x \cup \pi_{x_0})\} \cup \pi_t$$

together as a new class $\pi_{(t-1)_0}$ between π_{t-1} and π_t. After checking all elements of this partition of Π', we will get a new partition $\{\pi_0, \pi_{0_0}, \pi_1, \pi_{1_0}\pi_{2_0}, \pi_2, \ldots, \pi_{s-1}, \pi_{s_0}, \pi_s\}$. It is easy to show that this is a partition of Π, because α is the carrier of α' and there must be a vertex (i, T_i) in Q such that $T_i = \alpha$ by Definitions 1 and 2.

By this ordered partition, we can construct an execution β such that when all processes finish the execution, the reachable simplex is α'. Without loss of generality, we assume $|\pi_0| \neq 0$. Step 1. Let all the processes in π_0 access object CO^d firstly such that all processes have returns and return their inputs, denoted this collection of executions by β_0; Step 2. Let all the processes in π_{0_0} access object CO^d concurrently after π_0, such that none of them has return, which represents that all of them are crash, denoted this collection of executions by β_{0_0}; Step 3. Along with this ordered partition, do the similar steps as step 1 and step 2, denoted these ordered collections of executions by $\beta_{(j-1)_0}$, β_j for $1 < j \leq s$. Let

$$\beta = \beta_0 \beta_{0_0} \beta_1 \beta_{1_0} \cdots \beta_{s_0} \beta_s$$

be the composite of all executions, such that for each process in execution β_{x_0} there is no return, and for each process in execution β_x there must be a return. If for some π_i or π_{j0} is an empty set, it needs just omit the corresponding executions in β. As a result, α' is a reachable simplex under execution β.

For any face of α, we just take the composite of the subset of corresponding executions. Since α is arbitrary, it follows that $\mathbb{D}_d(\mathcal{I})$ is a subset of $\mathcal{P}^1(\mathcal{I})$. From upper arguments, we can see that $\mathcal{P}^1(\mathcal{I})$ is equal to $\mathbb{D}_d(\mathcal{I})$. □

Corollary 1. *When the input complex is $\mathcal{I}$, the protocol complex $\mathcal{P}^s(\mathcal{I})$ of any s-round full-information protocol in d-solo model is exactly the s-fold chromatic d-join $\mathbb{D}_d^s(\mathcal{I})$.*

Proof of Corollary 1. Given arbitrary execution ϵ of protocol $\mathcal{P}$ with s rounds, we can model ϵ as a sequence $\epsilon_1\epsilon_2\cdots\epsilon_s$ of executions, such that each process runs a protocol with at most one round in execution ϵ_i. Then each return of a process after executing ϵ_i is an input value of this process in execution ϵ_{i+1}, by Lemma 3 and Definition 1, $\mathcal{P}^s(\mathcal{I})$ is equal to the s-fold chromatic d-join $\mathbb{D}_d^s(\mathcal{I})$. □

Theorem 1. *A decision n-task $T = (\mathcal{I}, \mathcal{O}, \Delta)$ has a wait-free protocol in d-solo model if and only if there is an integer $k \geq 0$ and a color-preserving simplicial map $\mu : \mathbb{D}_d^k(\mathcal{I}) \longrightarrow \mathcal{O}$ such that for any simplex $\alpha \in \mathcal{I}$ and $\beta \in \mathbb{D}_d^k(\alpha)$, $\mu(\beta) \in \Delta(\alpha)$.*

Proof of Theorem 1. From the Figure 5, we know that the protocol $\mathcal{P}$ solves the task $(\mathcal{I}, \mathcal{O}, \Delta)$ iff there exists a simplicial color-preserving decision map $\delta : \mathcal{P}^k(\mathcal{I}) \longrightarrow \mathcal{O}$ such that for every simplex $S^r \in \mathcal{I}$, and every simplex $\widetilde{S}^r \in \mathcal{P}^k(S^r)$, $0 \leq r \leq n$, $\delta(\widetilde{S}^r) \in \Delta(S^r)$.

Suppose there exists a protocol that solves task T. By Corollary 1, the protocol complex is equal to $\mathbb{D}_d^k(\mathcal{I})$. Let $\mu = \delta$, which shows the necessity of the theorem.

For sufficiency, suppose S^r is any r-simplex in $\mathcal{I}$, apply algorithm as Table 1 in d-solo model and after k-round, any output simplex will agree on a simplex in $\mathbb{D}_d^k(S^r)$. By hypothesis, there exists a simplicial map $\mu : \mathbb{D}_d^k(S^r) \longrightarrow \mathcal{O}$. Then a process that chooses vertex v in $\mathbb{D}_d^k(S^r)$ then chooses as its output the value labeling $\mu(v)$. That is to say, this algorithm can solve this task. □

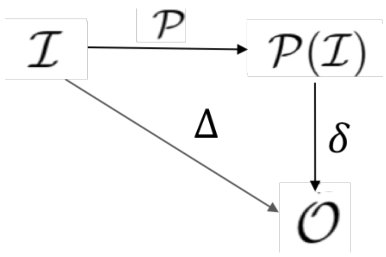

Figure 5. Commutative diagram.

Remark 3. *This theorem gives the topological characterization, he span that is a pair of a chromatic join and a color-preserving simplicial map, of the wait-free solvability of a given decision task in d-solo systems, which we can use to give us theoretical help in hardware design or early warning.*

Corollary 2. *Let $T = (\mathcal{I}, \mathcal{O}, \Delta)$ be a decidsion n-task, if T is not solvable in d_1-solo model then it is also unsolvable in d_2-solo model for $d_1 \leq d_2$.*

Proof of Corollary 2. Suppose T is solvable in d_2-solo model, then there is a color-preserving simplicial map μ from $\mathbb{D}_{d_2}^k(\mathcal{I})$ to $\mathcal{O}$ for some integer k by Theorem 1. Note that $\mathbb{D}_{d_1}^k(\mathcal{I})$ is a subcomplex of $\mathbb{D}_{d_2}^k(\mathcal{I})$ for $d_1 \leq d_2$, then the restriction of μ on $\mathbb{D}_{d_2}^k(\mathcal{I})$ is a carrier and color preserving simplicial map from $\mathbb{D}_{d_1}^k(\mathcal{I})$ to $\mathcal{O}$, by Theorem 1, T is solvable in d_1-solo model, which is a contradiction. □

3. Application

A standard input-less n-task $T = (\mathcal{I}, \mathcal{O}, \Delta)$ is a task of $n+1$ processes. Specifically, $\mathcal{I}$ is a pure complex having only one n-simplex, implying that each process has no input; $\mathcal{O}$ has a subcomplex L which is isomorphic to the boundary of an n-simplex. If less than $n+1$ processes participate in an execution, the output simplices must lie in L, otherwise, any simplex in $\mathcal{O}$ is eligible. For simplicity, we can use $T = (\mathcal{O}, L)$ to stand for $T = (\mathcal{I}, \mathcal{O}, \Delta)$. A standard input-less 2-task can be seen in Figure 3.

By Gafni and Koutsoupias in [23], we know that 2-task T is solvable in wait-free iterated immediately snapshot model if and only if L is contractible (i.e., there exists a continuous map from a 2-dimensional disk to $|\mathcal{O}|$ such that it maps the boundary of this disk to $|L|$) in $\mathcal{O}$ whose $n+1$-dimensional homotopy group can be represented. However, we will see that it is not true in d-solo model, that is to say, it is too weak to derive the solvability of task T from the condition in which L is just contractible in $\mathcal{O}$.

In this subsection, we mainly investigate the solvability of any standard input-less task, that is giving the necessary and sufficient conditions of solvability to these tasks, which also implies the classification of these tasks along their solvability in d-solo model. But before this, we need introduce the other topology space.

Intuitively, the standard chromatic subdivision of an n-simplex τ is an evolution of its barycentric subdivision, in which we need add the least new vertices in all but 0-dimensional faces of τ, such that each face is divided into more than one small pieces and each one is a chromatic simplex with the same colors and dimension of that face. However, in the next definition, we introduce a special chromatic subdivision in which the added vertices only appears in a face with its dimension at least k.

Definition 3. *Let α be an $[n]$-chromatic simplex and k be a non-negative integer. The k-type chromatic subdivision of α, denoted by $\mathbb{E}_k(\alpha)$, is a complex obtained from $\mathbb{D}_k(\alpha)$ by removing simplices having any vertex (c, T) with $1 \le dim(T) < k$.*

Remark 4. *If $n < k$, then $\mathbb{E}_k(\alpha) = \alpha$, that is, there is no any operation on α. If $k = 0, 1$ then $\mathbb{E}_k(\alpha)$ is equal to $Ch(\alpha)$ that is standard chromatic subdivision of α.*

We can inductively define the s-fold k-type chromatic subdivision $\mathbb{E}_k^s(\alpha)$ of α with that $\mathbb{E}_k^s(\alpha) = \mathbb{E}_k(\mathbb{E}_k^{s-1}(\alpha))$ for any positive integer s. Similarly, if X is an $[n]$-chromatic n-complex, we can get the s-fold k-type chromatic subdivision of X, denoted by $\mathbb{E}_k^s(X) \triangleq \bigcup_{\alpha \in X} \mathbb{E}_k^s(\alpha)$. In addition, if $s = 0$, set $\mathbb{E}_k^0(X) = X$. The Figure 6 is an example that shows 1- type and 2-type chromatic subdivisions of a 2-simplex.

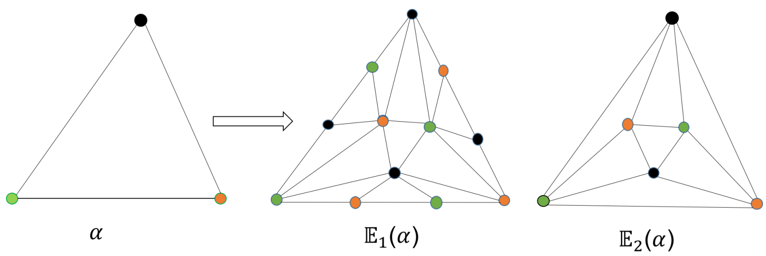

Figure 6. Once 1-type and 2-type chromatic subdivision of 2-simplex α.

By definition, $\mathbb{E}_k^s(X)$ is a subcomplex of $\mathbb{D}_k^s(X)$. Hence, if there is a simplicial map from $\mathbb{D}_k^s(X)$ to some chromatic complexes Y, there is a simplicial map from $\mathbb{E}_k^s(X)$ to Y for a given chromatic complexes Y. The next lemma implies the situation vice versa, which is usually not right for a pair of general simplicial complexes.

Lemma 4. *Let X and Y be $[n]$-chromatic n-simplexes. For any integers $k > 0$ and $s \geq 0$, there exists a color-preserving simplicial map from $\mathbb{D}_k^s(X)$ to $\mathbb{E}_k^s(Y)$.*

Proof of Lemma 4. We will show that by induction. Suppose $X = \{(i, v_i)\}_{i\in[n]}$ and $Y = \{(i, w_i)\}_{i\in[n]}$. When $s = 0$, $\mathbb{D}_k^0(X) = X$ and $\mathbb{E}_k^0(Y) = Y$, let $\psi_0 : X \longrightarrow Y$ that takes (i, v_i) to (i, w_i). When $s = 1$, assume $V(\mathbb{D}_k(X)) = \{(i_t, S_{r_t})\}_{t\in[n]}$ and $V(\mathbb{D}_k(Y)) = \{(i_t, S'_{r_t})\}_{t\in[n]}$, $i_t, r_t \in [n]$. Let ψ_1 be a point-to-point map from $V(\mathbb{D}_k(X))$ to $V(\mathbb{E}_k(Y)) \subseteq V(\mathbb{D}_k(Y))$ such that

$$\psi_1((i_t, S_{r_t})) = \begin{cases} (i_t, \psi_0((i_t, v_{i_t}))) & \text{if } dim(S_{r_t}) < k \\ & \text{and } (i_t, v_{i_t}) \in S_{r_t} \\ (i_t, S'_{r_t}) & \text{if } dim(S_{r_t}) \geq k \end{cases}$$

We will show that ψ_1 is a color-preserving simplicial map. Suppose τ is an arbitrary n-simplex in $\mathbb{D}_k(X)$, then it is a k-join, that is, it is compatible and $||L(\tau)|| \leq k$, then there are at most k vertices in τ such that $dim(S) < k$ for each vertex (c, S) of them. By former Equation, $\psi_1(\tau) \subseteq V(Ch(Y))$. Note that ψ_1 is color-preserving and there are at most k vertices in $\psi_1(\tau)$ such that $dim(\widetilde{S}) = 0$ for each vertex $(c, \widetilde{S})$ of them, then $\psi_1(\tau)$ is compatible and $||L(\psi_1(\tau))|| \leq k$, and then $\psi_1(\tau) \in \mathbb{D}_k(Y)$. Since each vertex of $\psi_1(\tau)$ has form $(i_t, (i_t, w_{i_t}))$ with $dim((i_t, w_{i_t})) = 0$, or (i_t, S'_{r_t}) with $dim(S'_{t_i}) \geq k$, then there is no vertex $(c, \widetilde{S})$ in $\psi_1(\tau)$ such that $1 < dim(X') < k$, and then $\psi_1(\tau) \in \mathbb{E}_k(Y)$. Since τ is arbitrary, it follows that ψ_1 is a color-preserving simplicial map.

Suppose that for $t < s$ there is a color-preserving simplicial map $\psi_t : \mathbb{D}_k^t(X) \longrightarrow \mathbb{E}_k^t(Y)$. Consider $t = s$. Let τ' be arbitrary n-simplex of $\mathbb{D}_k^{t'}(X)$, where $t' = s - 1 < s$. By assumption, there exists a color-preserving simplicial map $\psi_{t'}$ from $\mathbb{D}_k^{t'}(X)$ to $\mathbb{E}_k^{t'}(Y)$, then the restriction $\psi_{t'}|_{\tau'}$ of $\psi_{t'}$ on τ' is also a color-preserving simplicial map from τ' to $\mathbb{E}_k^{t'}(Y)$.

Since X and Y are arbitrary, set $X = \tau'$, $Y = \psi_{t'}|_{\tau'}(\tau')$, by the former argument, we know that there is a color-preserving simplicial map $g_{\tau'}$ from $\mathbb{D}_k(\tau')$ to $\mathbb{E}_k(\psi_{t'}|_{\tau'}(\tau'))$. As all such maps agree on their intersections, then we just constructed

$$g = \sqcup_{\tau'\in\mathbb{D}_k^{t'}(X)} g_{\tau'}$$

It follows that g is a color-preserving simplicial map from $\mathbb{D}_k^s(X)$ to $\mathbb{E}_k(\psi_{t'}(\mathbb{D}_k^{t'}(X)))$.

Note that $\psi_{t'}(\mathbb{D}_k^{t'}(X))$ is a subset of $\mathbb{E}_k^{t'}(Y)$, then $\mathbb{E}_k(\psi_{t'}(\mathbb{D}_k^{t'}(X)))$ is a subset of $\mathbb{E}_k^s(Y)$. Let I_0 be an inclusion map from $\mathbb{E}_k(\psi_{t'}(\mathbb{D}_k^{t'}(X)))$ to $\mathbb{E}_k^s(Y)$, then it is a color-preserving simplicial map. Let

$$\psi_s = I_0 \circ g$$

be the composite of I_0 and g, then ψ_s is the color-preserving simplicial map from $\mathbb{D}_k^s(X)$ to $\mathbb{E}_k^s(Y)$. By now, we complete the induction. □

Remark 5. *Let us explain what is the construction map ψ_1. Since X and Y are chromtatic n-simplexes, then $\mathbb{D}_k(X)$ is isomorphic to $\mathbb{D}_k(Y)$. In other words, they are the same in topology. Because $\mathbb{E}_k(Y)$ is a subcomplex of $\mathbb{D}_k(Y)$, then ψ_1 can be regarded as a collapse map from $\mathbb{D}_k(Y)$ to its subcomplex $\mathbb{E}_k(Y)$ essentially. That is to say ψ_1 map a simplex to itself if the simplex is in both $\mathbb{D}_k(Y)$ and $\mathbb{E}_k(Y)$, otherwise ψ_1 collapses the simplex to one of its vertices.*

Corollary 3. *Let X and Y be two arbitrary $[n]$-chromatic n-simplexes. If there is a color-preserving simplicial map from $\mathbb{E}_k^s(X)$ to Y for some non-negative integer s, there exists a color-preserving simplicial map from $\mathbb{D}_k^s(X)$ to Y.*

Proof of Corollary 3. Assume φ is the color-preserving simplicial map from $\mathbb{E}_k^s(X)$ to Y. By Lemma 4, there is a color-preserving simplicial map ψ from $\mathbb{D}_k^s(X)$ to $\mathbb{E}_k^s(X)$. Let

$$h = \varphi \circ \phi$$

be the composite of ϕ and φ, then h is a color-preserving simplicial map from $\mathbb{D}_k^s(X)$ to Y. $\square$

Suppose I^n be a $[n]$-chromatic simplex with $\partial I^n = L_0$. Let $\mathcal{F}_k^s$ be the collection of all color-preserving simplicial maps that are from $\mathbb{E}_k^s(I^n)$ to any $[n]$-chromatic complex X^n with a subcomplex L' which is isomorphic to L_0, such that $f(\mathbb{E}_k^s(L_0)) = L'$ for any $f \in \mathcal{F}_k^s$.

Definition 4. $\mathfrak{N}_k^s \triangleq \{f(\mathbb{E}_k^s(I^n)) | f \in \mathcal{F}_k^s\}$ *is called s-length k-nest, while* $\mathfrak{N}_k \triangleq \cup_{s=0}^{\infty} \mathfrak{N}_k^s$ *is called the k-nest.*

Intuitively, the k-nest $\mathfrak{N}_k$ is a space of images of all color-preserving simplicial maps that are from s-fold k-type chromatic subdivision of a chromatic n-simplex to any chromatic n-complex for all non-negative integer s.

We say that a chromatic n-complex K holds k-nest structure which is bounded by a subcomplex L' means that there exists a subcomplex K' of K with boundary L', such that K' is isomorphic to an element $M \in \mathfrak{N}_k$.

Observation For the spectral sequence about nests $\mathfrak{N}_0 = \mathfrak{N}_1, \mathfrak{N}_2, \dots, \mathfrak{N}_n$, if $1 \le i < j \le n$, $\mathfrak{N}_i$ is *larger* than $\mathfrak{N}_j$, that is, there is at least one element in $\mathfrak{N}_i$ but not in $\mathfrak{N}_j$.

In fact, we can always construct an element that is in $\mathfrak{N}_i$ but not in $\mathfrak{N}_j$ for $1 \le i < j \le n$. A simple example shows that the left complex of the Figure 7 is in $\mathfrak{N}_1$ but not in $\mathfrak{N}_2$.

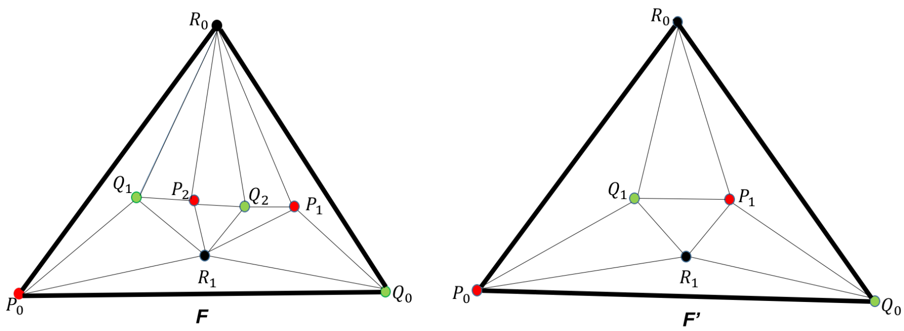

Figure 7. The complex in the left does not hold 2-nest structure, while the complex in the right holds 2-nest structure. But both of them hold 1-nest structure.

Example 3. *There are two chromatic* 2*-complexes in Figure 7 with color* $\{P, Q, R\}$*. The left one denoted by* F*, and the right one denoted by* F'*. Each vertex has form* C_t*, where* C *and* t *stand for color and some state of that vertex. The boundaries of them are the outer bold and black segments, denoted by* L*. There is no difficulty to show that* F' *holds* 2*-nest structure, because we only take* $s = 1$*,* $X = I^2$ *which is a chromatic* 2*-simplex with color* $\{P, Q, R\}$*, then there is a natural color-preserving simplicial from* $\mathbb{E}_2(I^2)$ *to* F'*. However, for* F*, we can never find an integer* s *such that there is a color-preserving simplicial from* $\mathbb{E}_2^s(I^2)$ *to* F*. But they all hold* 1*-nest structure, because,* L *is always contractible and* $\mathbb{E}_1$ *is the standard chromatic subdivision operation. Then there always exists a large enough integer* s *such that there is a color-preserving simplicial from* $\mathbb{E}_1^s(I^2)$ *to* F *and* F' *by Lemma 2 of [23].*

Theorem 2. *Let* $T = (\mathcal{O}, L)$ *be a standard input-less n-task. Then it is solvable in d-solo model if and only if* $\mathcal{O}$ *holds d-nest structure that is bounded by* L*.*

Proof of Theorem 2. For necessity. Assume T is solvable and I^n is an input n-simplex, then there is a color-preserving simplicial map $\mu : \mathbb{D}_d^s(I^n) \longrightarrow \mathcal{O}$, which maps $\mathbb{D}_d^s(\partial I^n)$ to L for some non-negative integer s by Theorem 1. Note that $\mathbb{E}_d^s(I^n)$ is a chromatic subcomplex of $\mathbb{D}_d^s(I^n)$, then the restriction of μ on $\mathbb{E}_d^s(I^n)$, denoted μ', is also a color-preserving simplicial map from $\mathbb{E}_d^s(I^n)$ to $\mathcal{O}$, which takes $\mathbb{E}_d^s(\partial I^n)$ to L. It follows that $\mu'(\mathbb{E}_d^s(I^n)) \in \mathfrak{N}_d$, which implies that $\mathcal{O}$ holds d-nest structure.

For sufficiency. Suppose $\mathcal{O}$ holds d-nest structure which is bounded by L, then there exists an integer s and a subcomplex $\mathcal{O}_0$ with boundary complex L, such that there is a color-preserving simplicial map f from $\mathbb{E}_d^s(I^n)$ to $\mathcal{O}_0$, which takes $\mathbb{E}_d^s(\partial I^n)$ to L. By Corollary 3, there is a color-preserving simplicial map ψ from $\mathbb{D}_d^s(I^n)$ to $\mathcal{O}_0$, which takes $\mathbb{D}_d^s(\partial I^n)$ to L. It follows that T is solvable by Theorem 1. □

Remark 6. *Even though Theorem 1 can be used to characterize the solvability of an input-less n-task in d-solo model, we find that it is difficult to describe the protocol complex and to find out the map δ. While, sometimes, we can check whether the output complex of the task holds d-nest structure or not easily. As a result, theorem 2 is an effective way to the characterization of solvability of tasks (at least for standard input-less task) in d-solo model.*

Corollary 4. *The power of computability of d-solo wait-free model is strictly getting weaker when d becomes bigger.*

Proof of Corollary. Consider standard input-less n-task $T = (\mathcal{O}, L)$. Let d_0, d_1 be two positive integers and $d_0 < d_1$. Note that d_1-nest structure $\mathfrak{N}_{d_1}$ does indeed a proper subset of d_0-nest structure $\mathfrak{N}_{d_0}$, then there does exist an element that is in $\mathfrak{N}_{d_0}$ but not in $\mathfrak{N}_{d_1}$. Assume $\mathcal{O}$ is that element, then $\mathcal{O}$ holds d_0-nest structure that is bounded by L but not holds d_1-nest structure. It follows that T is solvable in d_0-solo model but is not solvable in d_1-solo model by Theorem 2. As a result, the power of computation of d_1-solo is strictly weaker than the power of the computation of d_0-solo. □

4. Discussion

It seems that our result, Theorem 1, is just some minor improvements of previous studies of Herlihy M. et al. [6,7,13,24] when someone catches a glimpse of them. Actually, that is not the case. In [6,7], the Asynchronous Computability Theorem (ACT) characterizes the tasks that can be solved in share-memory models, that is to say, it is the case when $d = 1$. While in [13,24] ACT characterizes the tasks that can be solved in message-passing models, which is the case when $d = n + 1$. d-solo models, as a bridge that links share-memory models with message-passing models, which is introduced by Herlihy M. et al. [24], where they just characterized a kind of special tasks, colorless tasks, that can be wait-free solvable. However, all the previous results are based on one of the central results of topology which is the Simplicial Approximation Theorem [37]. That theorem establishes what is a "discrete version" of a continuous map. What a pity, this theorem cannot be used in a d-solo model when $d > 1$, because it is no longer the case that the diameter of the simplices in a subdivision is reduced. Our results bypass the Simplicial Approximation Theorem, and the span that is a pair of a protocol complex and a simplicial map δ is constructed based on the topological structure of the protocol complex itself. At the same times, our ACT will be appropriate not just for colorless tasks but for arbitrary decision tasks. From these views, our result is a great improvement.

In addition, the topological framework itself holds the potential applications in real life, such as the information safety [39,40], the public safety networks [41,42], the big data analytics [43,44] and so on. Let us take the information safety as an example. Consider the realistic scenario [45] of communication of information among a Trade Associations that consists of thousands of trading partners. The information, which is asynchronous, must be transmitted and delivered timely in an exchange process. One practical problem is that how can we use it to design an information exchange system which is as safe and efficient as possible [39,45]. Since we have obtained the computability of the d-solo models, we can choose the corresponding d according to the actual demand. The other is that how to do application development of a given information exchange system. Such as how to detect the defects of the system and make early warning [46]. Some traditional ways may cost large mount of time or money and sometimes a result hardly convinces many people. Luckily, it seems to be easy, at least on a theoretical level, if we use the topological framework to construct a mechanism and then to analyze it. From this point, our framework has a far-reaching impact on the production and living. Furthermore, there should be more and more researchers who take part in the research that how to apply topological framework to real-life better.

Lastly, the chromatic join of a given chromatic complex itself is very interesting and it provides a way to generate a new chromatic complex from the original complex. Though the geometrical properties of the generated chromatic complex look more complicated than the original, their topological properties are similar. Roughly, if an input complex is isomorphic to an n-sphere, then the chromatic join of an input complex is isomorphic to a wedge sum of n-spheres, and the bigger d the more number of n-spheres in the wedge sum. So our research has a potential value to the studies of topology itself, especially in simplicial homotopy theory.

5. Conclusions

In this work, we develop an elegant theory, the Asynchronous Computability Theorem in arbitrary solo systems when we focus on just full-information protocol and crash-failure models, which can be used to characterize the computability of a given decision task in that systems. Later, we derive a simple necessary and sufficient conditions for the solvability of input-less tasks in d-solo systems. It is a fly in the ointment that we maybe concentrate little on the topological properties of the k-folds chromatic join d-join $\mathbb{D}_d^k(\cdot)$ themselves and do not pay much attention to the general protocols and the byzantine models, because it seems that those properties are a little contribution to our main result, and the general protocols complexes or the byzantine models are quite complicated and are beyond the scope of our small paper. However, the byzantine failure model [9,17,18] and general protocol [47] play an important part in theoretical computer science, especially in distributed systems. So there are more meaningful research we should do in future work. Such as, how do we describe the protocol complexes of general protocols in d-solo systems clearly. What's the characterization of solvability of a decision tasks in that d-solo systems for the byzantine failure model? In addition, there is a little discussion on the practical application of our framework in this work. While the potential of applications and prospects are enormous, so there is much work we should do in that aspect in the future.

Author Contributions: Conceptualization, Y.Y. and X.L.; Methodology, F.L. and J.W.; Formal analysis, Y.Y. and X.L.; Writing—original draft preparation, Y.Y.; Writing—review and editing, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by Key-Area Research and Development Program of Guangdong Province under Grant 2020B010164003, and partially by the Natural Science Foundation of China under Grant 11971144 and High-level Scientific Research Foundation of Hebei Province, and partially by the National Natural Science Foundation of China under Grant 61702480, and partially by Youth Innovation Promotion Association of Chinese Academy of Sciences under Grant 2013073, and partially by State Key Laboratory of Computer Architecture Open Fund under Grant CARCH3410.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Fischer, M.J.; Lynch, N.A.; Paterson, M. Impossibility of Distributed Consensus with One Faulty Process. *J. ACM* **1985**, *32*, 374–382. [CrossRef]
2. Borowskym E.; Gafni, E. A Simple Algorithmically Reasoned Characterization of Wait-Free Computations (Extended Abstract). In Proceedings of the Sixteenth Annual ACM Symposium on Principles of Distributed Computing, Berlin, Germany, 20–22 June 1997; James, E., Burns, J.E., Attiya, H., Eds.; ACM: New York, NY, USA, 1997; pp. 189–198.

3. Chaudhuri, S. Agreement is Harder than Consensus: Set Consensus Problems in Totally Asynchronous Systems. In Proceedings of the Ninth Annual ACM Symposium on Principles of Distributed Computing, Quebec City, QC, Canada, 22–24 August 1990; Dwork, C., Ed.; ACM: New York, NY, USA, 1990; pp. 311–324.
4. Dolev, D.; Lynch, N.A.; Pinter, S.S.; Stark, E.W.; Weihl, W.E. Reaching approximate agreement in the presence of faults. *J. ACM* **1986**, *33*, 499–516. [CrossRef]
5. Saks, M.E.; Zaharoglou, F. Wait-free k-set agreement is impossible: The topology of public knowledge. *SIAM J. Comput.* **2000**, *29*, 1449–1483. [CrossRef]
6. Herlihy, M.; Shavit, N. The asynchronous computability theorem for t-resilient tasks. In Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing, San Diego, CA, USA, 16–18 May 1993; Kosaraju, S.R., Johnson, D.S., Aggarwal, A., Eds.; ACM: New York, NY, USA, 1993; pp. 111–120.
7. Herlihy, M.; Shavit, N. The topological structure of asynchronous computability. *J. ACM* **1999**, *46*, 858–923. [CrossRef]
8. Herlihy, M.; Rajsbaum, S. The topology of distributed adversaries. *Distrib. Comput.* **2013**, *26*, 173–192. [CrossRef]
9. Mendes, H.; Tasson, C.; Herlihy, M. Distributed computability in Byzantine asynchronous systems. In Proceedings of the Forty-Sixth annual ACM symposium on Theory of computing, New York, NY, USA, 1–3 June 2014; Shmoys, D.B., Ed.; ACM: New York, NY, USA, 2014; pp. 704–713.
10. Herlihy, M.; Rajsbaum, S.; Raynal, M.; Stainer, J. From wait-free to arbitrary concurrent solo executions in colorless distributed computing. *Theor. Comput. Sci.* **2017**, *683*, 1–21. [CrossRef]
11. Biran, O.; Moran, S.; Zaks, S. A Combinatorial Characterization of the Distributed Tasks Which Are Solvable in the Presence of One Faulty Processor. In Proceedings of the Seventh Annual ACM Symposium on Principles of Distributed Computing, Toronto, ON, Canada, 15–17 August 1988; Dolev, D., Ed.; ACM: New York, NY, USA, 1988; pp. 263–275.
12. Borowsky, E.; Gafni, E. Generalized FLP impossibility result for t-resilient asynchronous computations. In Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing, San Diego, CA, USA, 16–18 May 1993; Kosaraju, S.R., Johnson, D.S., Aggarwal, A., Eds.; ACM: New York, NY, USA, 1993; pp. 91–100.
13. Herlihy, M.; Rajsbaum, S.; Tuttle, M.R. Unifying Synchronous and Asynchronous Message-Passing Models. In Proceedings of the Seventeenth Annual ACM Symposium on Principles of Distributed Computing, Puerto Vallarta, Mexico, 28 June–2 July 1998; Coan, B.A., Afek, Y., Eds.; ACM: New York, NY, USA, 1998; pp. 133–142.
14. Junqueira, F.; Marzullo, K. A framework for the design of dependent-failure algorithms. *Concurr. Comput. Pract. Exp.* **2007**, *19*, 2255–2269. [CrossRef]
15. Moses, Y.; Rajsbaum, S. The Unified Structure of Consensus: A *Layered Analysis* Approach. In Proceedings of the Seventeenth Annual ACM Symposium on Principles of Distributed Computing, Puerto Vallarta, Mexico, 28 June–2 July 1998; Coan, B.A., Afek, Y., Eds.; ACM: New York, NY, USA, 1998; pp. 123–132.
16. Herlihy, M.; Rajsbaum, S. The topology of shared-memory adversaries. In Proceedings of the 29th Annual ACM Symposium on Principles of Distributed Computing, Zurich, Switzerland, 25–28 July 2010; Richa, A.W., Guerraoui, R., Eds.; ACM: New York, NY, USA, 2010; pp. 105–113.
17. Lamport, L.; Shostak, R.E.; Pease, M.C. The Byzantine Generals Problem. *ACM Trans. Program. Lang. Syst.* **1982**, *4*, 382–401. [CrossRef]
18. Braud-Santoni, N.; Guerraoui, R.; Huc, F. Fast byzantine agreement. In *PODC'13, Proceedings of the ACM Symposium on Principles of Distributed Computing, Montreal, QC, Canada, 22–24 July 2013*; Fatourou, P., Taubenfeld, G., Eds.; ACM: New York, NY, USA, 2013; pp. 57–64.
19. Herlihy, M.; Rajsbaum, S. The Decidability of Distributed Decision Tasks (Extended Abstract). In Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, TX, USA, 4–6 May 1997; Leighton, F.T., Shor, P.W., Eds.; ACM: New York, NY, USA, 1997; pp. 589–598.
20. Herlihy, M.; Rajsbaum, S. A classification of wait-free loop agreement tasks. *Theor. Comput. Sci.* **2003**, *291*, 55–77. [CrossRef]
21. Liu, X.W.; Xu, Z.W.; Pan, J.Z. Classifying rendezvous tasks of arbitrary dimension. *Theor. Comput. Sci.* **2009**, *410*, 2162–2173. [CrossRef]
22. Yue, Y.G.; Lei, F.C.; Wu, J. The evolution of non-degenerate and degenerate rendezvous tasks. *Topol. Appl.* **2019**, *264*, 187–200. [CrossRef]

23. Gafni, E.; Koutsoupias, E. Three-Processor Tasks Are Undecidable. *SIAM J. Comput.* **1999**, *28*, 970–983. [CrossRef]
24. Herlihy, M.; Rajsbaum, S.; Raynal, M.; Stainer, J. Computing in the Presence of Concurrent Solo Executions. In *Proceedings of the LATIN 2014: Theoretical Informatics-11th Latin American Symposium*; Pardo, A., Viola, A., Eds.; Springer: Berlin, Germany, 2014; Volume 8392, pp. 214–225.
25. Jung, D.; Cheng, G.; Zelinsky, A. Experiments in Realising Cooperation between Autonomous Mobile Robots. In *Experimental Robotics V, The Fifth International Symposium*; Casals, A., Almeida, A.T., Eds.; Springer: Berlin, Germany, 1997.
26. Prencipe, G. Instantaneous Actions vs. Full Asynchronicity: Controlling and Coordinating a Set of Autonomous Mobile Robots. In Proceedings of the Theoretical Computer Science, 7th Italian Conference, Torino, Italy, 4–6 October 2001; Antonio Restivo, A., Rocca, S.R.D., Roversi, L., Eds.; Springer: Berlin, Germany, 2001; pp. 185–190.
27. Suzuki, I.; Yamashita, M Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM J. Comput.* **1999**, *28*, 1347–1363. [CrossRef]
28. Cieliebak, M.; Prencipe, G. Gathering autonomous mobile robots. In Proceedings of the 9th International Colloquium on Structural Information and Communication Complexity, Andros, Greece, 10–12 June 2002; Christos Kaklamanis, C., Kirousis, L.M., Eds.; Carleton Scientific: Ottawa, ON, Canada, 2002; Volume 13, pp. 57–72.
29. Flochini, P.; Prencipe, G.; Santoro, N. *Distributed Computing by Oblivious Mobile Robots*; Morgan & Claypool: San Rafael, CA, USA, 2012.
30. Potop-Butucaru, M.; Raynal, M.; Tixeuil, S. Distributed computing with mobile robots: An introductory survey. In Proceedings of the 14th International Conference on Network-Based Information Systems, Tirana, Albania, 7–9 September 2011; Barolli, L.; Xhafa, F., Takizawa, M., Eds.; IEEE CS: Los Alametos, CA, USA, 2011; pp. 318–324.
31. Afek, Y.; Attiya, H.; Dolev, D.; Gafni, E.; Merritt, M.; Shavit, N. Atomic snapshots of shared memory. *J. ACM* **1993**, *40*, 873–890. [CrossRef]
32. Rajsbaum, S.; Castañeda, A.; Flores-Peñaloza, D.; Alcantara, M. Fault-tolerant robot gathering problems on graphs with arbitrary appearing times. In Proceedings of the 2017 IEEE International Parallel and Distributed Processing Symposium, Orlando, FL, USA, 29 May–2 June 2017; Warnow, T.J., Ed.; IEEE CS: Los Alametos, CA, USA, 2017; pp. 493–502.
33. Borowsky, E.; Gafni, E.; Lynch, N.A.; Rajsbaum, S. The BG distributed simulation algorithm. *Distrib. Comput.* **2001**, *14*, 127–146. [CrossRef]
34. Herlihy, M.; Kozlov, D.N.; Rajsbaum, S. *Distributed Computing Through Combinatorial Topology*; Morgan Kaufmann: San Francisco, CA, USA, 2013.
35. Hatcher, A. *Algebraic Topology*; CUP: Cambridge, UK, 2002.
36. Spanier, E.H. *Algebraic Topology*; Springer: Berlin, Germany, 1994.
37. Munkres, J.R. *Elements of Algebraic Topology*; Addison Wesley: Boston, MA, USA, 1984.
38. Kozlov, D. *Combinatorial Algebraic Topology*; Springer: Berlin, Germany, 2008.
39. Holly, S. Towards optimized exchange topologies in smart distribution grids. *Energy Inform.* **2018**, 45, 384–428. [CrossRef]
40. Yong, L.; Dan, Y. Information exchange in virtual communities under extreme disaster conditions. *Decis. Support Syst.* **2011**, *50*, 529–538.
41. Rabieekenari, L.; Sayrafian, K.; Baras, J.S. Autonomous relocation strategies for cells on wheels in environments with prohibited areas. In Proceedings of the 2017 IEEE International Conference on Communications, Paris, France, 21–25 May 2017; Gesbert, D., Debbah, M., Mellouk, A., Eds.; IEEE: Los Alamitos, CA, USA, 2017; pp. 1–6.
42. Tsiropoulou, E.E.; Koukas, K.; Papavassiliou, S. A Socio-physical and Mobility-Aware Coalition Formation Mechanism in Public Safety Networks. *EAI Endorsed Trans. Future Internet* **2018**, *4*, 154176. [CrossRef]
43. Stai, E.; Karyotis, V.; Katsinis, G.; Tsiropoulou, E.E.; Papavassilion, S. A Hyperbolic Big Data Analytics Framework within Complex and Social Networks. In *Big Data in Complex and Social Networks*; My, T., Wu, W., Xiong, H., Eds.; CRC: Boca Raton, FL, USA, 2016; pp. 3–36.

44. Florent, T. Big Data of Complex Networks and Data Protection Law: An Introduction to an Area of Mutual Conflict. In *Big Data of Complex Networks*; Dehmer, M., Emmert-Streib, F., Pickl, S., Holzinger, A., Eds.; CRC: Boca Raton, FL, USA, 2016; pp. 211–223.
45. Alison, J.K. Trade Associations as Information Exchange Mechanisms. *Rand J. Econ.* **1988**, *19*, 138–146.
46. Lazaros, K.G.; Maciej, K.; Nina, H.F. Anomaly detection through information sharing under different topologies. *EURASIP J. Inf. Secur.* **2017**, *2017*. [CrossRef]
47. Hoest, G.; Shavit, N. Towards a Topological Characterization of Asynchronous Complexity. In Proceedings of the Sixteenth Annual ACM Symposium on Principles of Distributed Computing, Santa Barbara, CA, USA, 21 August 1997; Burns, J.E.; Attiya, H., Eds.; ACM: New York, NY, USA, 1997; pp. 199–208.