Byzantine Approximate Agreement on Graphs

Thomas Nowak [ORCID]
Université Paris-Sud, France
Centre National de la Recherche Scientifique, Paris, France
thomas.nowak@lri.fr

Joel Rybicki [ORCID]
Institute of Science and Technology Austria, Klosterneuburg, Austria
joel.rybicki@ist.ac.at

──── **Abstract** ────

Consider a distributed system with n processors out of which f can be Byzantine faulty. In the approximate agreement task, each processor i receives an input value x_i and has to decide on an output value y_i such that

1. the output values are in the convex hull of the non-faulty processors' input values,
2. the output values are within distance d of each other.

Classically, the values are assumed to be from an m-dimensional Euclidean space, where $m \geq 1$.

In this work, we study the task in a discrete setting, where input values with some structure expressible as a graph. Namely, the input values are vertices of a finite graph G and the goal is to output vertices that are within distance d of each other in G, but still remain in the graph-induced convex hull of the input values. For $d = 0$, the task reduces to consensus and cannot be solved with a deterministic algorithm in an asynchronous system even with a single crash fault. For any $d \geq 1$, we show that the task is solvable in asynchronous systems when G is chordal and $n > (\omega + 1)f$, where ω is the clique number of G. In addition, we give the first Byzantine-tolerant algorithm for a variant of lattice agreement. For synchronous systems, we show tight resilience bounds for the exact variants of these and related tasks over a large class of combinatorial structures.

1 Introduction

In a distributed system, processors often need to coordinate their actions by jointly making consistent decisions or collectively agreeing on some data. While distributed systems can be resilient to failures, the extent to which they do so varies dramatically depending on the underlying communication and timing model, the fault model, and the level of coordination required by the task at hand. Exploring this interplay is at the core of distributed computing.

In this work, we investigate to which degree agreement can be reached in message-passing systems with Byzantine faults when (1) the set of input values has some discrete, combinatorial structure and (2) the set of output values must satisfy some structural closure property over the input values. We consider deterministic algorithms and assume a system with fully-connected point-to-point communication topology consisting of n processors out of which f may experience Byzantine failures, where the faulty processors may arbitrarily deviate from the protocol (e.g., crash, omit messages, or send malicious misinformation). We consider both asynchronous and synchronous systems. In the former, the processors do not have access to a shared global clock and sent messages may take arbitrarily long (but finite) time to be delivered. In the synchronous case, computation and communication proceeds in a lock-step fashion over discrete rounds.

1.1 Fault-tolerant distributed agreement tasks

Let P denote the set of n processors and $F \subseteq P$ some (unknown) set of faulty processors, where $|F| \leq f$. Many distributed agreement problems take the following form: Each processor $i \in P$ receives some input value $x_i \in V$, where V is the set of possible input values. The task is to have every non-faulty processor $i \in P \setminus F$ (irreversibly) decide on an output value $y_i \in V$ subject to some *agreement* and *validity* constraints. These constraints are commonly defined over the sets $X = \{x_i : i \in P \setminus F\}$ of input and $Y = \{y_i : i \in P \setminus F\}$ of output values of *non-faulty* processors. By choosing different constraints, one obtains different types of agreement problems.

1.1.1 Consensus and k-set agreement

Consensus is one of the most elementary problems in distributed computing [49]: all non-faulty processors should output a single value (agreement) that was the input of some non-faulty processor (validity). A natural generalisation of consensus is the k-set agreement problem [8], which is defined by the following constraints:

- agreement: $|Y| \leq k$ (all non-faulty processors decide on at most k values),
- validity: $Y \subseteq X$ (each decided value was an input of some non-faulty processor).

The special case $k = 1$ is the consensus problem and is known to be impossible to solve in an asynchronous setting even with $V = \{0, 1\}$ under a single crash fault using deterministic algorithms [28]. Analogously, k-set agreement cannot in general be solved in an asynchronous message-passing systems if there are $f \geq k$ crash faults [33, 6]. Note that for k-set agreement, it is natural to consider also other validity constraints [9].

1.1.2 Approximate agreement

While consensus and k-set agreement cannot in general be solved in an asynchronous system, it is however possible to obtain *approximate* agreement – in the sense that output values are close to each other – even in the presence of Byzantine faults. Formally, in the (multidimensional) approximate agreement problem, we are given $\varepsilon > 0$ and the set $V = \mathbb{R}^m$ of values forms an m-dimensional Euclidean space for some $m \geq 1$. The task is to satisfy

- agreement: $\text{dist}(y, y') \leq \varepsilon$ for any $y, y' \in Y$ (output values are within Euclidean distance ε),
- validity: the set Y is contained in the convex hull $\langle X \rangle$ of the set X of nonfaulty input values.

For an arbitrary $m \geq 1$, Mendes et al. [47] showed that under Byzantine faults the problem is solvable in asynchronous systems if and only if $n > (m + 2)f$ holds.

1.1.3 Lattice agreement

Lattice agreement is another well-studied relaxation of consensus with applications in renaming problems and obtaining atomic snapshots [3, 2, 23, 58]. In this problem, the set V of values forms a *semilattice* $\mathbb{L} = (V, \oplus)$, i.e., an idempotent commutative semigroup. The $\oplus$ operator defines a partial order $\leq$ over V defined as $u \leq v \iff u \oplus v = v$. The task is to decide on values that lie on a non-trivial chain, i.e., values that are comparable under $\leq$:

- agreement: $y \leq y'$ or $y' \leq y$ for any $y, y' \in Y$,
- validity: for any $y \in Y$ there exists some $x \in X$ such that $x \leq y$ and $y \leq \bigoplus X$.

Note that under crash faults the validity condition is usually given as $x_i \leq y_i \leq \bigoplus \{x_j : j \in P\}$ for $i \in P \setminus F$, which is less suitable in the context of Byzantine faults since otherwise output values could exit the convex hull defined by the correct processes' input values.

1.2 Structured agreement problems

Unlike the k-set agreement problem, the approximate and lattice agreement problems impose additional structure on the set V of values. In the former, the values form a (continuous) m-dimensional Euclidean space, whereas in the latter there is algebraic structure. Furthermore, the validity conditions require that the output respects some *closure property* on the input values. In approximate agreement, the closure is given by the convex hull operator in Euclidean spaces, whereas in lattice agreement, the output must reside in the minimal superset of X closed under $\oplus$.

Such closure systems have been studied under the notion of *abstract convexity spaces* and have a rich theory [36, 21, 18, 22, 55]. A (finite) convexity space on V is a collection $\mathcal{C}$ of subsets of V that satisfies

1. $\emptyset, V \in \mathcal{C}$,
2. $A, B \in \mathcal{C}$ implies $A \cap B \in \mathcal{C}$.

As the name suggests, the sets in $\mathcal{C}$ are called *convex* and every convexity space has the natural closure operator, which maps any set $A \in V$ to a minimal convex superset $A \subseteq \langle A \rangle \in \mathcal{C}$ called the convex hull of A. *Convex geometries* [21] are an important class of convexity spaces, which satisfy the Minkowski-Krein-Milman property: the closure $\langle A \rangle$ of any set $A \subseteq V$ is the closure of its *extreme points*, where $a \in A$ is an extreme point of A if $a \notin \langle A \setminus a \rangle$. Convex geometries have been studied extensively in a wide variety of *combinatorial* structures, such as graphs and hypergraphs [34, 24, 25, 19, 50, 48, 17], and partially ordered sets [18, 21, 51].

There has been extensive research on developing theory of convexity over *combinatorial* structures, such as graphs and hypergraphs [34, 24, 25, 19, 50, 48, 17], partially ordered sets [18, 21, 51], and so on. Much of the research has focused on identifying analogues to classical convexity invariants, such as Helly, Carathéodory, and Radon numbers, in various abstract convexity spaces [36, 34, 19, 20, 4, 17]. Convex geometries also have deep connections with matroid and antimatroid theory [11, 39]: convex geometries are duals of antimatroids, and a special class of greedoids, which provide a structural framework for characterising greedy algorithms [38, 39], are convex geometries [21]. Lovász and Saks [44] used theory of convex geometries to analyze a broad class of two-party communication complexity problems.

1.3 Approximate agreement on graphs

As our main example of an agreement problem with discrete, combinatorial structure, we focus on a problem where the set V of values has relational structure in the form of a connected graph $G = (V, E)$. In the *monophonic approximate agreement problem on G* the task is to output a set of vertices that satisfy
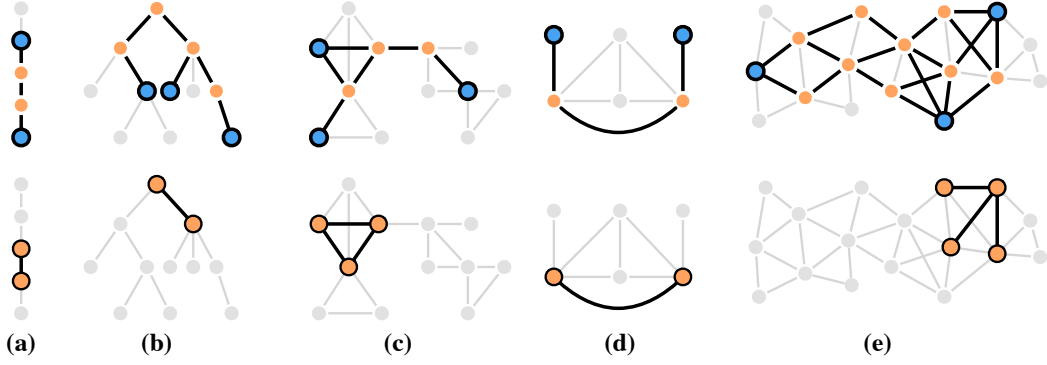
Figure 1 Examples of geodesic and monophonic agreement on graphs. In the top row, the blue and orange vertices form a convex hull of the blue vertices for each graph under (a)–(d) geodesic and (e) monophonic convexities. The thick edges lie in the shortest (geodesic) or chordless (monophonic) paths between the blue vertices. The bottom row shows possible feasible outputs for the respective approximate agreement problems with $d = 1$, i.e., the highlighted vertices form a clique (agreement) and are contained in the respective convex hull of the input values (validity).

- agreement: the set Y of output has diameter at most d for a given $d \geq 1$,
- validity: each value $y \in Y$ lies on a chordless[1] path between some input vertices $x, x' \in X$.

The above problem is a natural generalisation of approximate agreement onto graphs. It is easy to see that the discrete version of one-dimensional approximate agreement is just approximate agreement on a path (Figure 1a). If G is a tree or a block graph[2], then the task is to output vertices that lie on the minimal vertex set connecting all input vertices (Figure 1b–c).

In the parlance of abstract convexity theory [34, 24, 25, 19], the validity condition requires that the output lies in the *monophonic*, or *minimal path*, or *chordless path* convex hull of the input vertices. Another reasonable validity constraint would be to require the output values to lie on the *shortest* paths between input vertices, i.e., in the *geodesic* convex hull. We consider both variants and refer to the latter version of the problem as *geodesic* approximate agreement on G.

1.4   Contributions

In this work, we introduce the *abstract approximate agreement problem on a convexity space* $\mathcal{C}$ satisfying:

- agreement: Y is a free set, that is, $\langle Y \rangle = \mathrm{ex}\, Y$, where $\mathrm{ex}\, Y$ is the extreme points of Y.
- validity: $Y \subseteq \langle X \rangle$.

While our primary focus lies in the graphical version of approximate agreement, we believe the abstract problem is also interesting in itself. Indeed, it conveniently turns out that the problem coincides with various natural agreement problems: In graphs, the monophonic and geodesic approximate agreement on graphs problem given above boils down to solving approximate agreement on the chordless path or geodesic convexities of G. Moreover, lattice agreement on $\mathbb{L}$ is equivalent to solving approximate agreement on the *algebraic convexity space* of the semilattice (sets closed under $\oplus$). Our key results can be summarised as follows:

---

[1] A path is *chordless* (also known as *minimal*) if there are no edges between non-consecutive vertices.
[2] A graph is a *block graph* if every 2-connected component is a clique.

1. **Byzantine approximate agreement on chordal graphs.** We give algorithms for approximate agreement on trees and chordal graphs. The algorithms tolerate $f < n/(\omega+1)$ Byzantine faults and terminate in $O(\log N)$ *asynchronous* rounds, where ω is the clique number and N is the number of vertices in the value graph G. In trees, we achieve optimal resilience.

2. **Byzantine lattice agreement on cycle-free semilattices.** As another example, we give an asynchronous lattice agreement algorithm on cycle-free lattices that tolerates up to $f < n/(\omega + 1)$ Byzantine faults, where ω is the height of the semilattice. To our knowledge, this is the first algorithm that solves any variant of semilattice agreement under Byzantine faults.

3. **General impossibility results for asynchronous systems.** We give impossibility results for approximate agreement on arbitrary convex geometries parameterised by two combinatorial convexity invariants: the Carathéodory number c and the Helly number ω. As corollaries, we obtain resilience lower bounds for approximate agreement problems in asynchronous systems.

4. **Optimal synchronous algorithms for convex consensus.** We consider the *exact* variant of the abstract approximate agreement problem, where the agreement constraint is replaced by $|Y| = 1$. While the problem cannot be solved in asynchronous systems, we show that it can be solved on any convex geometry $\mathcal{C}$ in $\Theta(f)$ synchronous rounds if and only if $n > \omega f$ holds, where ω is the Helly number of $\mathcal{C}$. Moreover, the upper bound holds for *any* convexity space.

Our work can be seen as an extension of the Mendes–Herlihy approximate agreement and Vaidya–Garg multidimensional consensus frameworks [46, 54, 47] onto general convexity spaces. However, while these operate in continuous m-dimensional Euclidean spaces, our analysis relies on combinatorial theory of abstract convexity, where the input and output values have discrete, combinatorial structure. In particular, the discrete nature of the convexity space poses new challenges, as unlike in the continuous setting, non-trivial convex sets do not necessarily contain non-extreme points to choose from to facilitate convergence.

Multidimensional agreement problems in Euclidean spaces have applications ranging from, e.g., robot convergence tasks to distributed voting and convex optimisation [47]. Our work extends the scope of these techniques to discrete convexity spaces, which can be used to describe various natural combinatorial systems. Finally, unlike prior work, our algorithms do not assume that processors can perform computations or send messages involving arbitrary precision real values, as in the discrete case a single value can be encoded using $O(\log |V|)$ bits.

1.5 Related work

The seminal result of Fischer et al. [28] showed that *exact* consensus cannot be reached in asynchronous systems in the presence of crash faults. Dolev et al. [14] showed that it is however possible to reach *approximate agreement* in an asynchronous system even with arbitrary faulty behavior when the values reside on the continuous real line. Subsequently, the one-dimensional approximate agreement problem has been extensively studied [14, 26, 27, 1]. Fekete [27] showed that any algorithm reducing the distance of values from d to ε requires $\Omega(\log(\varepsilon/d))$ asynchronous rounds when $f \in \Theta(n)$; in the discrete setting this yields the bound $\Omega(\log N)$ for paths of length N. Recently, Mendes et al. [47] introduced the natural generalisation of *multidimensional* approximate agreement and showed that the m-dimensional problem is solvable in an asynchronous system with Byzantine faults if and only if $n > (m + 2)f$ holds for any given $m \geq 1$.

The *lattice agreement problem* was originally introduced in the context of wait-free algorithms in shared memory models [3, 2]. The problem has recently resurfaced in the context of asynchronous message-passing models with crash faults [23, 58]. These papers consider the problem when the validity condition is given as $x_i \leq y_i \leq \bigoplus\{x_j : j \in P\}$, i.e., the output of a processor must satisfy $x_i \leq y_i$ and the feasible area is determined also by the inputs of faulty processors. However, it is not difficult to see that under Byzantine faults, this validity condition is not reasonable, as the problem cannot be solved even with one faulty processor.

Another class of structured agreement problems in the wait-free asynchronous setting are *loop agreement* tasks [32], which generalise k-set agreement and approximate agreement (e.g., $(3, 2)$-set agreement and one-dimensional approximate agreement). In loop agreement, the set of inputs consists of three distinct vertices on a loop in a 2-dimensional simplicial complex and the outputs are vertices of the complex with certain constraints, whereas *rendezvous tasks* are a generalisation of loop agreement to higher dimensions [43]. These tasks are part of large body of work exploring the deep connection of asynchronous computability and combinatorial topology, which has successfully been used to characterise the *solvability* of various distributed tasks [31]. Gafni and Kuznetsov's P-reconciliation task [29] achieves geodesic approximate agreement on a graph of system configurations.

Finally, we note that distributed agreement tasks play a key role in many fault-tolerant clock synchronisation algorithms [57, 42, 41]. Byzantine-tolerant clock synchronisation can be solved using one-dimensional approximate agreement [57], whereas in the *self-stabilising* setting both exact digital clock synchronisation [41] and pulse synchronisation tasks reduce to consensus [42]. However, while the latter problem has been extensively studied [16, 13, 42], non-trivial lower bounds are still lacking [42]. Given that clock synchronisation closely relates to agreement on cyclic structures, investigating agreement tasks on different structures may yield insight into the complexity of fault-tolerant (approximate) clock synchronisation. Indeed, we show that agreement on graphs *without* long induced cycles is considerably easier than consensus.

2   Preliminaries

We start with some basic preliminaries needed to describe the main ideas and results of the paper.

2.1   Abstract convexity spaces

Let V be a finite set. The collection $\mathcal{C} \subseteq 2^V$ is a *convexity space on V* if (1) $\emptyset, V \in \mathcal{C}$ holds, and (2) $A, B \in \mathcal{C}$ implies that $A \cap B \in \mathcal{C}$. A set $K \in \mathcal{C}$ is said to be *convex*. For any $A \subseteq V$, the *convex hull* of A is the minimal convex set $\langle A \rangle \in \mathcal{C}$ such that $A \subseteq \langle A \rangle$. Thus, $\langle \cdot \rangle$ is a closure operator on V. For any $A \subseteq V$ and $a \in A$, a is called an *extreme point* of A if $a \notin \langle A \setminus a \rangle$. For a convex set $K \in \mathcal{C}$, we use ex K to denote the extreme points of K. The convexity space $\mathcal{C}$ is a *convex geometry* if every $K \in \mathcal{C}$ satisfies $K = \langle \text{ex } K \rangle$. A convex set K is *free* if $K = \text{ex } K$. Finally, a nonempty set $A \subseteq V$ is *irredundant* if $\partial A \neq \emptyset$ where $\partial A = \langle A \rangle \setminus \bigcup_{a \in A} \langle A \setminus a \rangle$. The following theorem characterises convex geometries:

▶ **Theorem 1** ([21]). *Let $\mathcal{C}$ be a convexity space on V. The following conditions are equivalent:*
1. *$\mathcal{C}$ is a convex geometry.*
2. *For every $K \in \mathcal{C}$, $K = \langle \text{ex } K \rangle$ (Minkowski-Krein-Milman property).*
3. *For every $K \in \mathcal{C} \setminus \{V\}$, there exists an element $u \in V \setminus K$ such that $K \cup \{u\} \in \mathcal{C}$.*

2.1.1 Carathéodory and Helly numbers

The *Carathéodory number* of a convexity space $\mathcal{C}$ on V is the smallest integer c such that for any $U \subseteq V$ and any $u \in \langle U \rangle$, there is a set $S \subseteq U$ such that $|S| \leq c$ and $u \in \langle S \rangle$. The Carathéodory number of a convexity space equals the maximum size of an irredundant set in $\mathcal{C}$. A collection $\mathcal{C}$ of sets is k-intersecting if every $\mathcal{B} \subseteq \mathcal{C}$ with $|\mathcal{B}| \leq k$ has a nonempty intersection. The *Helly number* of a convexity space $\mathcal{C}$ is the smallest integer ω such that any finite ω-intersecting $\mathcal{A} \subseteq \mathcal{C}$ has a nonempty intersection. If $\mathcal{C}$ is a convex geometry, the Helly number equals the maximum cardinality of a free set in $\mathcal{C}$ [21].

2.1.2 Examples of convex geometries

In this work, we focus on the following convexity spaces:

- Let $G = (V, E)$ be a graph. A set $U \subseteq V$ is convex if all the vertices on all minimal, i.e., chordless, paths connecting any $u, v \in U$ are contained in U. The Helly number of this convexity space equals the size of the maximum clique in G [34, 19] and the Carathéodory number is at most two [19]. Moreover, free sets coincide with cliques. The convexity space is a convex geometry iff G is chordal [24]. Indeed, if G contains an induced cycle K of length at least four, then it is easy to check that K is convex, but has no extreme points.
- Let $\mathbb{L} = (V, \oplus)$ be a semilattice and $\mathcal{C}$ be the collection of sets closed under $\oplus$. The collection $\mathcal{C}$ is a convex geometry, where every $K \in \mathcal{C}$ corresponds to a subsemilattice $(K, \oplus)$ of $\mathbb{L}$. A set K is free if and only if it is a chain [51]. Thus, the Helly number of a semilattice equals its height. Moreover, the Carathéodory number equals the breadth of the semilattice.

2.2 Asynchronous rounds

When operating in the asynchronous model, we describe and analyse the algorithms in the *asynchronous round* model. In this model, each processor has a local round counter and labels all of its messages with a round number. Each correct node initialises its round counter to 0 at the start of the execution and increases its local round counter from t to $t + 1$ only when it has received at least $n - f$ messages belonging to round t (since up to f faulty nodes may omit their messages). In each round $t \geq 0$, a non-faulty processor i

1. sends a value to each processor $j \in P$,

2. receives a value $M_{ij}(t)$ from each processor $j \in P$,

3. updates local state and proceeds to round $t + 1$.

The received message $M_{ij}(t)$ may be empty, denoted by $\perp$, to indicate that no message arrived from processor j (e.g., due to a crash or a delay). We use the set

$$P_i(t) = \{j \in P : M_{ij}(t) \neq \perp\}$$

to denote the processors from which i received a nonempty message on round t.

Assuming $n > 3f$ and with the help of reliable broadcast, the witness technique [1, 47], and attaching round numbers to all messages, the Byzantine asynchronous round model can guarantee the following for each $i, j \in P \setminus F$:

1. $|P_i(t)| \geq n - f$,

2. $|P_i(t) \cap P_j(t)| \geq n - f$,

3. if $M_{ik}(t) = x \neq \perp$ for $k \in F$, then $M_{jk}(t) \in \{x, \perp\}$.

That is, (1) every correct processor receives at least $n - f$ nonempty values (out of which f may be from faulty processors), (2) any two correct processors receive at least $n - f$ common values (possibly f of which may be from faulty processors), and (3) if some correct processor receives a nonempty value x from a faulty processor, then all other correct processors receive the same value or no value. The Byzantine asynchronous round model can be simulated in the asynchronous model so that a non-faulty processor broadcasts $O(n \log n)$ additional bits per round [1, 47].

2.3 Graphs

Let $G = (V, E)$ be a finite undirected connected graph, where $V = V(G)$ denotes the set of vertices and $E = E(G)$ the set of edges. We assume all graphs are simple (no parallel edges) and loopless (no self-loops). For any $U \subseteq V$, we use $G[U] = (U, F)$, where $F = \{\{u, v\} \in E : u, v \in U\}$, to denote the subgraph of G induced by the vertices in U. An ℓ-length path $u \rightsquigarrow v$ from a vertex u to vertex v is a non-repeating sequence $(u = v_0, \ldots, v_\ell = v)$ of vertices such that $\{v_i, v_{i+1}\} \in E$. An ℓ-cycle is an $(\ell - 1)$-path from u to v with $\{u, v\} \in E$. A path $v_0, \ldots, v_\ell$ is *chordless* (or minimal) if there does not exist any edge $\{v_i, v_j\} \in E$ for $j > i + 1$.

For vertices $u, v \in V$, we denote the length of the shortest path between u and v as $d(u, v)$. The eccentricity of vertex v is $\mathrm{ecc}(v) = \max\{d(v, u) : u \in V(G)\}$. For a set $U \subseteq V$, we define its diameter $D(U) = \max\{d(u, v) : u, v \in U\}$. The diameter of graph G is denoted by $D(G) = D(V)$. The *radius* of a graph G is $R(G) = \min\{\mathrm{ecc}_G(v) : v \in V(G)\}$ and the *center* is $\mathrm{center}(G) = \{u \in V(G) : \mathrm{ecc}_G(u) = R(G)\} \subseteq V(G)$. For a connected set $U \subseteq V$, we use the short-hands $R(U) = R(G[U])$ and $\mathrm{center}(U) = \mathrm{center}(G[U])$.

A graph G is a *tree* if it contains no cycles and *chordal* if contains no ℓ-cycle with $\ell \geq 4$ as an induced subgraph. A graph is *Ptolemaic* if it is chordal and distance-hereditary (any connected induced subgraph preserves distances). The clique number $\omega(G)$ is the size of the largest clique in G. A vertex is *simplicial* in $U \subseteq V$ if its neighbourhood $\mathcal{N}(v) \cap U$ in U is a clique. A perfect elimination ordering $\preceq$ of $G = (V, E)$ is a total order on V such that any $u \in V$ is simplicial in $\{v : u \preceq v\}$. A graph has a perfect elimination ordering iff it is chordal.

2.3.1 Chordal graphs

Chordal graphs (also known as triangulated, rigid or decomposable graphs) form an important and well-studied class of graphs. From a structural point of view, they have many equivalent characterisations: they are graphs that have no induced cycles greater than three, graphs for which perfect elimination orderings exist, graphs in which every minimal vertex separator is a clique, and others [12, 52, 30, 53, 24].

Due to their ubiquitous nature and convenient structural properties, the algorithmic aspects of chordal graphs have received much attention in the past decades. For example, chordal graphs have applications in a variety of contexts including combinatorial and semi-definite optimisation [56] and probabilistic graphical models [40]. Indeed, many NP-hard problems, such as finding maximum cliques or optimal vertex colourings, often admit simple polynomial time solutions in chordal graphs [30]. In the distributed setting, it is possible to find good approximations to minimum vertex colourings and maximum independent sets in chordal graphs [37].

2.4 Lattices

A (join) semilattice is an idempotent commutative semigroup $\mathbb{L} = (V, \oplus)$, where V is a finite set and $\oplus : V \times V \to V$ is called the *join* operator. A semilattice has a natural partial order defined as $u \leq v \iff u \oplus v = v$, where $u \oplus v$ is the least upper bound (i.e., a join) of $\{u, v\}$ in the partial order. We write $u < v$ if $u \leq v$ and $u \neq v$. For any set $U = \{u_0, \ldots, u_\ell\} \subseteq V$, the least upper bound $\bigoplus U$ is known as the join of U. If $u_0 \leq \ldots \leq u_\ell$ holds, then U is said to be a chain from u_0 to u_ℓ. If $\{u, v\}$ is a chain, then u and v are said to be comparable. The height $\omega(\mathbb{L})$ of a semilattice is the maximum cardinality of any chain $U \subseteq V$. The breadth of a semilattice is the smallest integer b such that for any nonempty $U \subseteq V$ there is a subset $A \subseteq U$ of size at most b satisfying $\bigoplus A = \bigoplus U$.

3 Approximate agreement on abstract convexity spaces

3.1 Iterative algorithm on abstract convexity spaces

In this section, we describe a basic step for an approximate agreement algorithm in the Byzantine asynchronous round model in an abstract convexity space $\mathcal{C}$ with Helly number ω. The algorithm is a generalisation of the Mendes–Herlihy algorithm by Mendes et al. [47] onto abstract convexity spaces. It is not guaranteed, however, to converge on *all* discrete convexity spaces.

The algorithm proceeds iteratively. At the start of each asynchronous round t, each correct processor $i \in P \setminus F$ broadcasts its current value $x_i(t) \in V$. At the end of round t, processor i has received a value from at least $n - f$ processors $P_i(t) \subseteq P$. These values are used to compute a new value $x_i(t+1) = y_i(t)$. For brevity, we often omit t from our notation, e.g., use the short-hands such as $P_i = P_i(t)$.

3.1.1 Computing the safe area

For any subset of processors $J \subseteq P_i$, define

$$\mathcal{V}_i(J) = \{M_{ij}(t) \neq \bot : j \in J\}$$

to be the set of values processor i received from processors in J. Processor i locally computes

$$\mathcal{K}_i = \left\{ \langle \mathcal{V}_i(J) \rangle : J \in \binom{P_i}{|P_i| - f} \right\} \quad \text{and} \quad H_i = \bigcap \mathcal{K}_i,$$

where $\langle \cdot \rangle$ denotes the convex hull operator. Processor i then outputs the value

$$y_i = \begin{cases} \phi(H_i) & \text{if } H_i \neq \emptyset, \\ \bot & \text{otherwise,} \end{cases}$$

where $\phi \colon \mathcal{C} \to V$ is an output map, which will depend on the convexity space $\mathcal{C}$ we are working in, see Section 4 for an output map for chordal graphs. The Helly property guarantees that H_i and y_i remain in the closure $\langle X \rangle$ of the input values. For each $t \geq 0$, we define $X(t) = \{x_j(t) : j \in P \setminus F\} \subseteq V$ and $Y(t) = \{y_j(t) = \phi(H_j(t)) : j \in P \setminus F\}$.

▶ **Lemma 2.** *Suppose $\mathcal{C}$ is a convexity space on V with Helly number ω. If $n > \max\{(\omega + 1)f, 3f\}$ holds, then for each iteration $t \geq 0$ the above algorithm satisfies*
- $\emptyset \neq H_i(t) \subseteq \langle X(t) \rangle$ *for all $i \in P \setminus F$,*
- $\bigcap_{i \in P \setminus F} H_i(t) \neq \emptyset$.

3.2   On the elimination of extreme points

If we can in each iteration remove some extreme point of $\langle X \rangle$, where X is the set of input values, then the hull of output values $\langle Y \rangle$ shrinks. In an arbitrary convexity space, a convex set may not have any extreme points (consider, e.g., the chordless path convexity on a four cycle). However, in a convex geometry every nonempty convex set has an extreme point, as $\langle X \rangle = \langle \text{ex}\, X \rangle$ by Theorem 1.

Moreover, finite convex geometries are in a sense "easy to peel" iteratively – at least from the global perspective. We say that a total order $\preceq$ on V is a *convex elimination order* of $\mathcal{C}$ if for any $u \in V$ the sets $A(u) = \{v \in V : u \preceq v\}$ and $A(u) \setminus u$ are convex. Theorem 1 implies that convex geometries admit such orderings and we assume that the values in V are labelled according to such an order $\preceq$. For a set $U \subseteq V$, we let $\max U$ and $\min U$ be the unique maximal and minimal elements, respectively, given by $\preceq$ such that for all $v \in U$ we have $\min U \preceq v$ and $v \preceq \max U$.

▶ Remark 3. For any $K \subseteq V$ it holds that $\min K \in \text{ex}\, K$ and $\min K = \min \langle K \rangle$.

The next lemma shows that to guarantee progress by shrinking the set of output values, it suffices to always exclude, e.g., $\min X$ from the output (of course, this is not indefinitely possible).

▶ **Lemma 4.** *If* $\min X \notin Y$ *in a convex elimination order, then* $\langle Y \rangle \subsetneq \langle X \rangle$.

4   Approximate agreement on chordal graphs

We now show that monophonic approximate agreement on chordal graphs can be solved given that $n > (\omega + 1)f$ holds, where ω is the clique number of G. This also implies that geodesic approximate agreement is solvable on Ptolemaic graphs. Throughout we assume that $G = (V, E)$ is a connected chordal graph with at least two vertices and $\mathcal{C}$ is its chordless path convexity space. We recall that the Helly number of $\mathcal{C}$ coincides with the clique number $\omega = \omega(G)$ of G [34, 19].

4.1   Approximate agreement on trees

Suppose $G = (V, E)$ is a tree. As G is also chordal, it has a perfect elimination ordering $\preceq$. We assume that the vertices of V are labelled according to this ordering and define the output map

$$\phi(K) = \max \text{center}\, K,$$

where $\text{center}\, K \subseteq V$ is the center of the subgraph $G[K]$ induced by K. This rule roughly divides the diameter of the set of active values by two, which yields the following result.

▶ **Theorem 5.** *If* $n > 3f$ *and* $G = (V, E)$ *is a tree, then approximate agreement on* G *can be solved in* $\log D(G) + 1$ *asynchronous rounds, where* $D(G)$ *is the diameter of* G.

4.2   Fast monophonic approximate agreement on chordal graphs

We now present a fast monophonic approximate agreement algorithm on chordal graphs. To this end, we use the tree algorithm above on a suitable tree decomposition of the actual graph G.
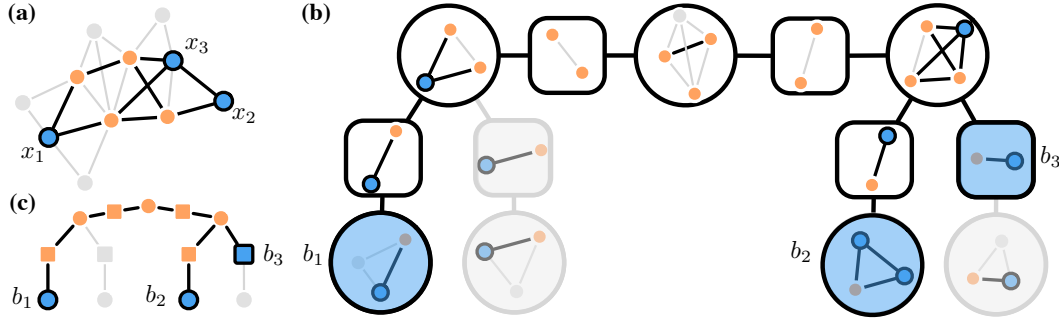
Figure 2 Approximate agreement on chordal graphs via clique trees. (a) The chordal value graph G and three input values $X = \{x_1, x_2, x_3\} \subseteq V(G)$. (b) An expanded clique tree (T, χ) of G. The bags $B = \{b_1, b_2, b_3\}$ satisfy $x_i \in \chi(b_i)$ and the bags are used as input for the approximate agreement algorithm on trees. The round bags are maximal cliques and the rectangular bags are the intersection of its neighbouring round bags, which are the minimal vertex separators in G. Note that a bag $b \in \langle B \rangle$ may contain vertices of G outside the convex hull $\langle X \rangle$ of the initial input values X (e.g., bag b_1 and the central bag). (c) The graph T on which we run the tree algorithm.

▶ **Definition 6.** *Let G be a graph, T a tree and $\chi \colon V(T) \to 2^{V(G)}$ be a mapping. We say that the pair (T, χ) is a* tree decomposition *of G if the following conditions are satisfied:*
- *for all $v \in V(G)$ there exists $b \in V(T)$ such that $v \in \chi(b)$,*
- *for all $e \in E(G)$ there exists $b \in V(T)$ such that $e \subseteq \chi(b)$,*
- *if $v \in \chi(a) \cap \chi(b)$, then $v \in \chi(c)$ for all $c \in V(T)$ residing on the unique path $a \rightsquigarrow b$.*

The tree decomposition is a clique tree *if each $b \in V(T)$ induces a maximal clique $\chi(b)$ in G.*

Chordal graphs can be characterised as those graphs having a clique tree [10, Proposition 12.3.11]. In fact, if G is chordal, the tree T can always be chosen as a spanning tree of G's clique graph, i.e., the graph whose nodes are the maximal cliques of G and whose edges join those cliques with nonempty intersection [7, Theorem 3.2]. Unlike non-chordal graphs in which the number of maximal cliques can be exponential, the number of maximal cliques is at most linear in chordal graphs:

▶ **Lemma 7** ([5]). *If G is a chordal graph, then it has a clique tree (T, χ) with $|V(T)| \in O(|V(G)|)$.*

For the purposes of our algorithm, we use a special kind of clique trees, which we call *expanded*. A clique tree (T, χ) is *expanded* if for each $\{a, b\} \in E(T)$ we have either $\chi(a) \subseteq \chi(b)$ or $\chi(b) \subseteq \chi(a)$; see Figure 2a–b for an example of an expanded clique tree.

▶ **Lemma 8.** *Every chordal graph G has an expanded clique tree T with $|V(T)| = O(|V(G)|)$.*

4.2.1 The algorithm

Let $G = (V, E)$ denote the chordal value graph, (T, χ) an expanded clique tree of G, and **A** the approximate agreement algorithm on trees given by Theorem 5. Given an input $x_i(0) \in V(G)$ on the graph G, processor $i \in P \setminus F$ starts by choosing any *bag* $b_i(0) \in V(T)$ such that the $x_i(0) \in b_i(0)$; see Figure 2. In iteration $t \geq 1$, every processor $i \in P \setminus F$ performs the following:

1. Broadcast $x_i(t)$ and $b_i(t)$ to all other processors.
2. Simulate one step of **A** on the $b(\cdot)$ values and set $b_i(t + 1) = \mathbf{A}(b_{0,i}(t), \ldots, b_{n-1,i}(t))$.
3. Compute the safe area H_i^G from the received values $x_{ij}(t)$.
4. Set $x_i(t + 1)$ to an arbitrarily chosen element of $\chi(b_i(t + 1)) \cap H_i^G$.
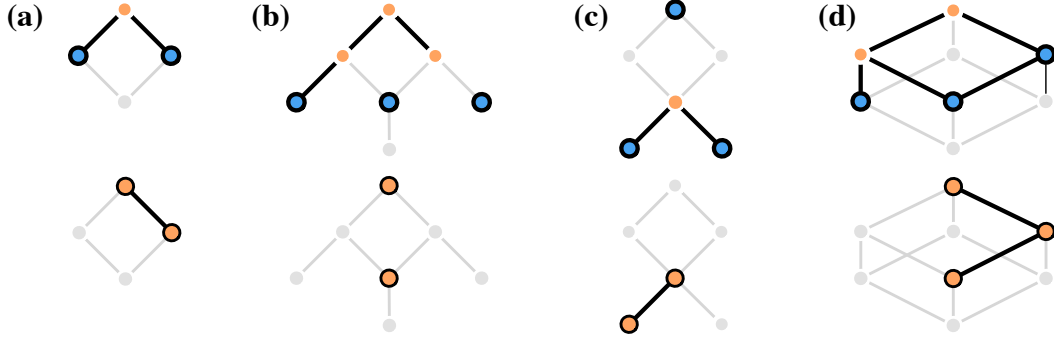
■ **Figure 3** Examples of algebraic convex sets on semilattices. The figures show the Hasse diagrams of these semilattices. In the top row, the blue vertices are input values and the orange vertices are contained in the hull of the blue vertices. The bottom row shows feasible outputs for these cases (i.e. chains contained in the convex hull). The semilattices (a)–(b) are cycle-free, whereas (c) and (d) respectively contain an induced 4- and 6-cycle in the comparability graph.

Since the $b_i(\cdot)$ values are updated using the algorithm **A**, these values converge onto a single edge $\{a, b\} \in E(T)$ in the tree T. As $\chi(a) \cup \chi(b)$ is a clique due to the expandedness of T, the output values $x(\cdot)$ will have diameter at most one in G assuming that $x_i(t+1)$ is well-defined for each $i \in P_i \setminus F$, that is, $\chi(b_i(t+1)) \cap H_i^G \neq \emptyset$. Showing this is the main challenge of the correctness proof.

▶ **Theorem 9.** *Let $G = (V, E)$ be a chordal graph. If $n > (\omega(G) + 1)f$, then monophonic approximate agreement on G can be solved in $O(\log|V|)$ asynchronous rounds.*

Finally, we observe that the above implies that *geodesic* approximate agreement can be solved in Ptolemaic graphs, as geodesic and monophonic convexities are identical on these graphs [24].

▶ **Corollary 10.** *If $G = (V, E)$ is a connected Ptolemaic graph and $n > (\omega(G) + 1)f$, then geodesic approximate agreement on G is solvable in $O(\log |V|)$ asynchronous rounds.*

5 Byzantine lattice agreement on cycle-free semilattices

The abstract convex geometry framework can be easily applied to solve agreement problems on other combinatorial structures. As an example we consider asynchronous Byzantine lattice agreement on a special class of semilattices. Let $\mathbb{L} = (V, \oplus)$ be a semilattice and $\leq$ its natural partial order. The *comparability graph* of $\leq$ is the graph $G = (V, E)$ where $\{u, v\} \in E$ if $u \neq v$ and u and v are comparable. A partial order $\leq$ is *cycle-free* if the comparability graph is chordal [45]. Similarly, we say $\mathbb{L}$ is cycle-free if $\leq$ is cycle-free. See Figure 3 for examples of cycle-free and non-cycle-free semilattices.

▶ **Lemma 11.** *Let $\mathbb{L} = (V, \bigoplus)$ be a cycle-free semilattice. There exists an elimination order $\preceq$ on the algebraic convexity of $\mathbb{L}$ such that $A(u) = \{v : u \oplus v \in \{u, v\}, u \preceq v\}$ is a chain for any $u \in V$.*

We assume now that $\preceq$ is the ordering given by Lemma 11 and use the following output map

$$\phi(K) = \begin{cases} \bigoplus K & \text{if } K \neq \text{ex } K \\ \max K & \text{otherwise.} \end{cases}$$

With this, the framework given in Section 3 and Lemma 4 yield the following result.

▶ **Theorem 12.** *Suppose $\mathbb{L} = (V, \oplus)$ is a cycle-free semilattice of height ω and $n > (\omega + 1)f$. Then Byzantine semilattice agreement on $\mathbb{L}$ can be solved in the asynchronous model.*

6 Resilience lower bounds for abstract convexity spaces

We obtain general lower bounds for asynchronous approximate agreement on abstract convexity spaces. We derived a general way of obtaining impossibility results using a partitioning argument and so-called blocking instances. This makes it possible to show how to obtain blocking instances for convex geometries from irredundant and free sets. Recall that the Carathéodory number c of $\mathcal{C}$ equals the maximum cardinality of an irredundant set in $\mathcal{C}$. This yields the following result, which can be seen as a generalisation of the Mendes et al. [47] lower bound technique from Euclidean spaces into arbitrary convexity spaces:

▶ **Theorem 13.** *Let $\mathcal{C}$ be a convexity space with Carathéodory number c and Helly number ω. Then:*

- *If $n \leq (c+1)f$, then there is no asynchronous abstract approximate agreement algorithm on $\mathcal{C}$ that on all inputs satisfies validity and agreement (i.e., the set of outputs is free).*
- *If $\mathcal{C}$ is a convex geometry and $n \leq (\omega + 1)f$, then there is no asynchronous abstract approximate agreement algorithm on $\mathcal{C}$ satisfying validity that outputs at most $\omega - 1$ distinct values.*

Combining the above result with classic results in combinatorial convexity theory gives lower bounds for specific problems. For any graph G with diameter at least two, the Carathéodory number is two [19] and clique is a free set. This implies the following result.

▶ **Corollary 14.** *The monophonic approximate agreement problem on any G with diameter at least two cannot be solved if $n \leq 3f$. There is no asynchronous algorithm that outputs a clique of size less than ω unless $n \leq (\omega + 1)f$.*

The case of Byzantine semilattice agreement is perhaps more interesting, as the breadth of a semilattice coincides with its Carathéodory number [35]. For any $b > 1$, there are semilattices with height and breadth equal to b: take the subsemilattice of a subset lattice over $[b]$ without $\emptyset$.

▶ **Corollary 15.** *Suppose $\mathbb{L}$ is a semilattice with breadth b. If $n \leq (b+1)f$, then there exists no asynchronous algorithm that solves Byzantine semilattice agreement on $\mathbb{L}$.*

7 Synchronous convex agreement

Finally, we give matching upper and lower bound results for exact convex consensus problem on abstract convexity spaces. It is easy to see that convex consensus is at least as hard as binary consensus, i.e., classic impossibility results for binary consensus [49, 28, 15] also hold for convex consensus. Hence, we consider the synchronous model of computation.

▶ **Theorem 16.** *Let $\mathcal{C}$ be an abstract convexity space on V with Helly number ω. If $n > \max\{3f, \omega f\}$, then convex consensus on $\mathcal{C}$ can be solved in $O(f)$ synchronous communication rounds using $O(nf^2)$ messages of size $O(n \cdot (\log n + \log |V|))$.*

It turns out that the higher resilience threshold of $n > \omega f$ for convex consensus is necessary already in the case of convex geometries.

▶ **Theorem 17.** *Let $\mathcal{C}$ be a convex geometry with a Helly number ω. If $n \leq \omega f$ holds, then convex consensus on $\mathcal{C}$ cannot be solved in the synchronous message-passing model.*

8 Conclusions

Many structured agreement tasks correspond to exact or approximate agreement problems on (possibly discrete) convexity spaces. Using the theory of abstract convexity, we have obtained Byzantine-tolerant algorithms for a large class of agreement problems on discrete combinatorial structures. In the synchronous model, exact convex consensus for any convexity space can be solved in an optimally resilient manner with asymptotically optimal round complexity. However, in the asynchronous setting, several interesting open problems remain.

1. It seems difficult to come up with a general rule for the output map $\phi : \mathcal{C} \to V$ in a way that guarantees that the convex hull of active values shrinks. Nevertheless, we have seen that on chordal graphs and cycle-free semilattices we can solve approximate agreement efficiently. In both cases, the underlying convexity space is a convex geometry. Given that the literature is abound with convex geometries associated with combinatorial structures [34, 24, 25, 19, 50, 48, 17, 18, 21, 51], it is natural to ask whether the abstract approximate agreement problem can be solved for other convex geometries as well.

2. It is unclear whether the abstract approximate agreement problem can be solved on general convexity spaces. For example, the asynchronous algorithms for approximate agreement on graphs presented here fail for non-chordal graphs: already the simplest example of a non-chordal graph, the four cycle, is difficult to handle. Indeed, the monophonic convexity of a four cycle is *not* a convex geometry: a convex set may not necessarily have any extreme points, and thus, greedily excluding extreme points does not seem to work. Are there resilient asynchronous algorithms that solve the problem for non-chordal graphs?

3. We obtained resilience lower bounds in terms of the Carathéodory and the Helly numbers. However, our positive results for the asynchronous model hold in cases where the Carathéodory number is at most two. Interestingly, in the continuous setting of multidimensional approximate agreement [47], tight resilience bounds exist, as the Carathéodory and Helly numbers coincide in the usual Euclidean convexity space on $\mathbb{R}^m$. Is there a discrete convexity space with a higher Carathéodory number in which approximate agreement can be solved?

References

1 Ittai Abraham, Yonatan Amit, and Danny Dolev. Optimal Resilience Asynchronous Approximate Agreement. In *Proc. International Conference on Principles of Distributed Systems (OPODIS 2015)*, pages 229–239, 2005. `doi:10.1007/11516798_17`.

2 Hagit Attiya and Arie Fouren. Adaptive and efficient algorithms for lattice agreement and renaming. *SIAM Journal on Computing*, 31(2):642–664, 2001.

3 Hagit Attiya, Maurice Herlihy, and Ophir Rachman. Atomic snapshots using lattice agreement. *Distributed Computing*, 8(3):121–132, 1995.

4 Rommel M Barbosa, Erika M. M. Coelho, Mitre C. Dourado, Dieter Rautenbach, and Jayme L. Szwarcfiter. On the Carathéodory number for the convexity of paths of order three. *SIAM Journal on Discrete Mathematics*, 26(3):929–939, 2012.

5 Anne Berry and Romain Pogorelcnik. A simple algorithm to generate the minimal separators and the maximal cliques of a chordal graph. *Information Processing Letters*, 111:508–511, 2011.

6 Martin Biely, Peter Robinson, and Ulrich Schmid. Easy impossibility proofs for *k*-set agreement in message passing systems. In *Proc. International Conference on Principles of Distributed Systems (OPODIS 2011)*, pages 299–312. Springer, 2011.

7   Jean R. S. Blair and Barry Peyton. An Introduction to Chordal Graphs and Clique Trees. In Alan George, John R. Gilbert, and Joseph W. H. Liu, editors, *Graph Theory and Sparse Matrix Computation*, pages 1–29. Springer, Heidelberg, 1993.

8   Soma Chaudhuri. More choices allow more faults: Set consensus problems in totally asynchronous systems. *Information and Computation*, 105(1):132–158, 1993.

9   Roberto De Prisco, Dahlia Malkhi, and Michael Reiter. On *k*-set consensus problems in asynchronous systems. *IEEE Transactions on Parallel and Distributed Systems*, 12(1):7–21, 2001.

10   Reinhard Diestel. *Graph Theory*. Springer, Heidelberg, 4th edition, 2010.

11   Brenda L. Dietrich. Matroids and antimatroids – a survey. *Discrete Mathematics*, 78(3):223–237, 1989.

12   Gabriel Andrew Dirac. On rigid circuit graphs. In *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, volume 25, pages 71–76. Springer, 1961.

13   Danny Dolev, Matthias Függer, Christoph Lenzen, and Ulrich Schmid. Fault-tolerant algorithms for tick-generation in asynchronous logic. *Journal of the ACM*, 61(5):30:1–30:74, 2014. `doi:10.1145/2560561`.

14   Danny Dolev, Nancy A. Lynch, Shlomit S. Pinter, Eugene W. Stark, and William E. Weihl. Reaching Approximate Agreement in the Presence of Faults. *Journal of the ACM*, 33(3):499–516, May 1986. `doi:10.1145/5925.5931`.

15   Danny Dolev and Rüdiger Reischuk. Bounds on information exchange for Byzantine agreement. *Journal of the ACM*, 32(1):191–204, 1985. `doi:10.1145/2455.214112`.

16   Shlomi Dolev and Jennifer L. Welch. Self-stabilizing clock synchronization in the presence of Byzantine faults. *Journal of the ACM*, 51(5):780–799, 2004. `doi:10.1145/1017460.1017463`.

17   Mitre C. Dourado, Dieter Rautenbach, Vinícius Fernandes dos Santos, Philipp M. Schäfer, and Jayme L. Szwarcfiter. On the Carathéodory number of interval and graph convexities. *Theoretical Computer Science*, 510:127–135, 2013. `doi:10.1016/j.tcs.2013.09.004`.

18   Pierre Duchet. Convexity in combinatorial structures. In *Proc. Winter School on Abstract Analysis*, pages 261–293. Circolo Matematico di Palermo, 1987.

19   Pierre Duchet. Convex sets in graphs, II. Minimal path convexity. *Journal of Combinatorial Theory, Series B*, 44(3):307–316, 1988. `doi:10.1016/0095-8956(88)90039-1`.

20   Jürgen Eckhoff. Helly, Radon, and Carathéodory type theorems. In *Handbook of Convex Geometry, Part A*, pages 389–448. Elsevier Science Publishers B.V, 1993.

21   Paul H. Edelman and Robert E. Jamison. The theory of convex geometries. *Geometriae Dedicata*, 19(3):247–270, 1985. `doi:10.1007/BF00149365`.

22   Paul H. Edelman and Michael E. Saks. Combinatorial representation and convex dimension of convex geometries. *Order*, 5(1):23–32, 1988. `doi:10.1007/BF00143895`.

23   Jose M. Faleiro, Sriram Rajamani, Kaushik Rajan, G. Ramalingam, and Kapil Vaswani. Generalized lattice agreement. In *Proc. ACM Symposium on Principles of Distributed Computing (PODC 2012)*, pages 125–134. ACM, 2012. `doi:10.1145/2332432.2332458`.

24   M. Farber and R. E. Jamison. Convexity in graphs and hypergraphs. *SIAM Journal on Algebraic Discrete Methods*, 7(3):433–444, 1986.

25   Martin Farber and Robert E. Jamison. On local convexity in graphs. *Discrete Mathematics*, 66(3):231–247, 1987. `doi:10.1016/0012-365X(87)90099-9`.

26   Alan David Fekete. Asymptotically optimal algorithms for approximate agreement. *Distributed Computing*, 4(1):9–29, 1990.

27   Alan David Fekete. Asynchronous approximate agreement. *Information and Computation*, 115(1):95–124, 1994.

28   Michael J. Fischer, Nancy A. Lynch, and Michael S. Paterson. Impossibility of distributed consensus with one faulty process. *Journal of the ACM*, 32(2):374–382, 1985. `doi:10.1145/3149.214121`.

29 Eli Gafni and Petr Kuznetsov. N-consensus is the second strongest object for $N+1$ processes. In *Proc. International Conference on Principles of Distributed Systems (OPODIS 2007)*, pages 260–273. Springer, Heidelberg, 2007.

30 Fănică Gavril. Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph. *SIAM Journal on Computing*, 1(2):180–187, 1972.

31 Maurice Herlihy, Dmitry Kozlov, and Sergio Rajsbaum. *Distributed Computing Through Combinatorial Topology*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2013.

32 Maurice Herlihy and Sergio Rajsbaum. A classification of wait-free loop agreement tasks. *Theoretical Computer Science*, 291(1):55–77, 2003.

33 Maurice Herlihy, Sergio Rajsbaum, and Mark R. Tuttle. Unifying synchronous and asynchronous message-passing models. In *Proc. ACM Symposium on Principles of Distributed Computing (PODC 1998)*, pages 133–142. ACM, 1998.

34 Robert E. Jamison and Richard Nowakowski. A Helly theorem for convexity in graphs. *Discrete Mathematics*, 51(1):35–39, 1984. `doi:10.1016/0012-365X(84)90021-9`.

35 Robert E Jamison-Waldner. A perspective on abstract convexity: classifying alignments by varieties. *Convexity and Related Combinatorial Geometry, New York*, 1982.

36 David Kay and Eugene W Womble. Axiomatic convexity theory and relationships between the Carathéodory, Helly, and Radon numbers. *Pacific Journal of Mathematics*, 38(2):471–485, 1971.

37 Christian Konrad and Viktor Zamaraev. Brief Announcement: Distributed Minimum Vertex Coloring and Maximum Independent Set in Chordal Graphs. In *Proc. ACM Symposium on Principles of Distributed Computing (PODC 2018)*, pages 159–161, New York, NY, USA, 2018. ACM. `doi:10.1145/3212734.3212787`.

38 Bernhard Korte and László Lovász. Greedoids – A Structural Framework for the Greedy Algorithm. In *Progress in Combinatorial Optimization*, pages 221–243. Academic Press, 1984. `doi:10.1016/B978-0-12-566780-7.50019-2`.

39 Bernhard Korte, László Lovász, and Rainer Schrader. *Greedoids*, volume 4. Springer Science & Business Media, 2012.

40 Steffen L. Lauritzen. *Graphical models*, volume 17. Clarendon Press, 1996.

41 Christoph Lenzen and Joel Rybicki. Near-optimal self-stabilising counting and firing squads. *Distributed Computing*, 2018. `doi:10.1007/s00446-018-0342-6`.

42 Christoph Lenzen and Joel Rybicki. Self-Stabilising Byzantine Clock Synchronisation is Almost as Easy as Consensus. *Journal of the ACM*, 2019. To appear.

43 Xingwu Liu, Zhiwei Xu, and Jianzhong Pan. Classifying rendezvous tasks of arbitrary dimension. *Theoretical Computer Science*, 410(21):2162–2173, 2009. `doi:10.1016/j.tcs.2009.01.033`.

44 László Lovász and Michael Saks. Communication complexity and combinatorial lattice theory. *Journal of Computer and System Sciences*, 47(2):322–349, 1993.

45 Tze-Heng Ma and Jeremy P. Spinrad. Cycle-free partial orders and chordal comparability graphs. *Order*, 8(1):49–61, March 1991. `doi:10.1007/BF00385814`.

46 Hammurabi Mendes and Maurice Herlihy. Multidimensional approximate agreement in Byzantine asynchronous systems. *Proc. Annual ACM symposium on Symposium on Theory of Computing (STOC 2013)*, page 391, 2013. `doi:10.1145/2488608.2488657`.

47 Hammurabi Mendes, Maurice Herlihy, Nitin Vaidya, and Vijay K. Garg. Multidimensional agreement in Byzantine systems. *Distributed Computing*, 28:423–441, 2015.

48 Morten H. Nielsen and Ortrud R. Oellermann. Steiner trees and convex geometries. *SIAM Journal on Discrete Mathematics*, 23(2):680–693, 2009.

49 Marshall C. Pease, Robert E. Shostak, and Leslie Lamport. Reaching agreement in the presence of faults. *Journal of the ACM*, 27(2):228–234, 1980. `doi:10.1145/322186.322188`.

50 Ignacio M. Pelayo. *Geodesic Convexity in Graphs.* Springer, 2013. `doi:10.1007/978-1-4614-8699-2`.

51 Paul Poncet. Convexities on ordered structures have their Krein–Milman theorem. *Journal of Convex Analysis*, 21(1):89–120, 2014.

52 Donald J. Rose. Triangulated graphs and the elimination process. *Journal of Mathematical Analysis and Applications*, 32(3):597–609, 1970.

53 Donald J. Rose, R. Endre Tarjan, and George S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on computing*, 5(2):266–283, 1976.

54 Nitin H. Vaidya and Vijay K. Garg. Byzantine vector consensus in complete graphs. In *Proc. ACM Symposium on Principles of Distributed Computing (PODC 2013)*, pages 65–73, 2013.

55 Marcel LJ van De Vel. *Theory of convex structures*, volume 50. Elsevier, 1993.

56 Lieven Vandenberghe and Martin S Andersen. Chordal graphs and semidefinite optimization. *Foundations and Trends in Optimization*, 1(4):241–433, 2015.

57 Jennifer Lundelius Welch and Nancy Lynch. A new fault-tolerant algorithm for clock synchronization. *Information and Computation*, 77(1):1–36, 1988.

58 Xiong Zheng, Changyong Hu, and Vijay K. Garg. Lattice Agreement in Message Passing Systems. In *Proc. International Symposium on Distributed Computing (DISC 2018)*, volume 121 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 41:1–41:17, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. `doi:10.4230/LIPIcs.DISC.2018.41`.