Combinatorial Control Barrier Functions: Nested Boolean and p-choose-r Compositions of Safety Constraints

Pio Ong, Haejoon Lee, Tamas G. Molnar, Dimitra Panagou, Aaron D. Ames

***Abstract*— This paper investigates the problem of composing multiple control barrier functions (CBFs)—and matrix control barrier functions (MCBFs)—through logical and combinatorial operations. Standard CBF formulations naturally enable conjunctive (AND) combinations, but disjunctive (OR) and more general logical structures introduce nonsmoothness and possibly a combinatorial blow-up in the number of logical combinations. We introduce the framework of *combinatorial CBFs* that addresses p-choose-r safety specifications and their nested composition. The proposed framework ensures safety for the exact safe set in a scalable way, using the original number of primitive constraints. We establish theoretical guarantees on safety under these compositions, and we demonstrate their use on a patrolling problem in a multi-agent system.**

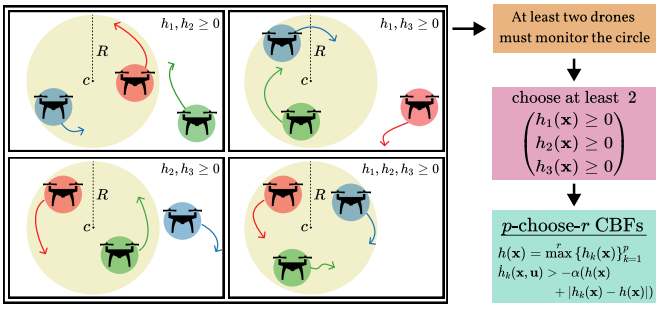

Fig. 1. An example of a p-choose-r constraint in multi-drone surveillance.

I. Introduction

Ensuring dynamic safety in modern control systems has become essential in applications such as robotics, autonomous vehicles, and aerospace systems. Control barrier functions (CBFs) [1] provide one of the most widely used tools for addressing safety. They were arguably popularized through their ability to be framed as safety filters using the quadratic program (QP) formulation [2]. The resulting optimization-based controllers facilitate the combination of multiple Lyapunov and barrier constraints to handle stability and safety simultaneously. This divide-and-conquer approach has been a key to the practical adoption of CBFs, making it straightforward to handle multiple control criteria.

As safety has taken on greater importance, many works, such as [3]–[5], have extended the QP formulation to integrate multiple CBFs simultaneously. While the optimization framework facilitates such integration, it also requires a verification on compatibility: even when each CBF admits a safeguarding control, there may not be one in conjunction. This issue has motivated studies on compatibility of multiple CBFs [6] and methods for ensuring feasibility of controllers with multiple CBFs [7], [8]. In certain cases, such as parallel safe set boundaries or box constraints, compatibility can be guaranteed [9], [10]. Nevertheless, the majority of these works remains limited to simple conjunctive combinations of CBFs, whereas practical systems often require richer logical structures among safety constraints.

PO and AA are with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125, USA. {pioong, ames}@caltech.edu.

HL and DP are with the Department of Robotics, University of Michigan, Ann Arbor, MI 48109, USA. {haejoonl, dpanagou}@umich.edu.

TM is with the Department of Mechanical Engineering, Wichita State University, Wichita, KS 67260, USA. tamas.molnar@wichita.edu.

This research was in parts supported by the Technology Innovation Institute and AFOSR Award #113535-19668.

Prior work on the composition of CBFs has largely focused on combining them into a single function and treating the result as a new CBF. For instance, the works [5], [11] perform AND/OR composition of CBFs through min/max functions and deal directly with the resulting nonsmooth barrier function. This formulation has been applied to marine vehicle applications [12]. Since nonsmooth functions are difficult to handle in general, other works approximate them with smooth functions such as softmin/softmax [13]–[15], enabling applications to quadrotors [16] and safe reinforcement learning [17]. This approach has been later extended to handle hierarchical complex objectives [18]. Another line of work combines CBFs through signal temporal logic [19], [20]. More recently, matrix-valued CBFs [21] have been proposed as a way to capture logical combinations. Despite this progress, the existing literature remains focused on Boolean (AND/OR) combinations of safety constraints. In contrast, here we consider the more general case of combinatorial compositions, safety specifications that commonly arise in fault-tolerant and multi-agent systems (see, e.g., Fig. 1). Our approach builds on the matrix CBF perspective of using multiple inequality constraints rather than a single one. We summarize our contributions next.

In this paper, we go beyond the Boolean compositions studied in prior works and develop a framework for the combinatorial composition of CBFs. We introduce the notion of p-choose-r CBFs that address safety problems where at least r out of p constraints must hold. These CBFs are defined based on sorting individual primitive constraints and they serve as pivots in the barrier conditions. Like in the standard AND case, our method enforces safety through multiple inequalities involving the primitive CBFs, which allows us to avoid nonsmoothness issues present in formulations based on nonsmooth min/max operators. In addition, the sorting and pivoting arguments extend naturally to nested

logical structures, while ensuring that the number of barrier conditions remains equal to the number of primitive CBFs, despite the combinatorial nature of the safe set composition. Moreover, unlike approaches using smooth relaxation, our method preserves the safe set exactly as specified by the logical combinations. We demonstrate the flexibility and scalability of the framework on a multi-agent patrolling task.

II. Background

A. Control barrier functions

Consider the control-affine system[1]:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} \tag{1}$$

with state $\mathbf{x} \in \mathbb{R}^n$ and control input $\mathbf{u} \in \mathbb{R}^m$. The system drift $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ and the control matrix $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^{m\times n}$ are assumed continuous. Then, if the control signal $t \mapsto \mathbf{u}(t)$ is continuous, there exists a continuously differentiable solution $t \mapsto \mathbf{x}(t)$ to the system (1). We are interested in ensuring any solution $\mathbf{x}(t)$ evolves within a safety constraint $\mathcal{C} \subset \mathbb{R}^n$.

In simpler problems, the safety constraint is given by a single continuously differentiable scalar function $h : \mathbb{R}^n \to \mathbb{R}$:

$$\mathcal{C} = \big\{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\big\}. \tag{2}$$

Then, one may address safety using control barrier functions.

Definition 1. *(Control Barrier Function, [2]):* A continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ is a *control barrier function* (CBF) for (1) if there exists $\alpha \in \mathcal{K}^{\mathrm{e}}$ such that, for each $\mathbf{x}$ in the set $\mathcal{C}$ in (2), there exists $\mathbf{u} \in \mathbb{R}^m$ satisfying:

$$\dot{h}(\mathbf{x}, \mathbf{u}) \triangleq L_{\mathbf{f}}h(\mathbf{x}) + \sum_{i=1}^{m} L_{\mathbf{g}_i}h(\mathbf{x})\mathbf{u}_i > -\alpha(h(\mathbf{x})). \tag{3}$$

When h is a CBF, we may find a control signal that keeps the set $\mathcal{C}$ *safe*, such that $\mathbf{x}(0) \in \mathcal{C} \implies \mathbf{x}(t) \in \mathcal{C}$ for all time, addressing our safety problem[2]. One way to design a safe control signal is via optimization. Given p CBFs, denoted by $\{h_k\}_{k=1}^p$, the corresponding constraints can be addressed simultaneously with an optimization-based controller:

$$\begin{aligned} \mathbf{k}(\mathbf{x}) = \underset{\mathbf{u}\in\mathbb{R}^m}{\operatorname{argmin}} \quad & \|\mathbf{u} - \mathbf{k}_{\mathrm{d}}(\mathbf{x})\|^2 \\ \text{s.t.} \quad & \dot{h}_k(\mathbf{x}, \mathbf{u}) \geq -\alpha(h_k(\mathbf{x})),\ \forall k \in [p], \end{aligned} \tag{4}$$

that changes a desired controller $\mathbf{k}_{\mathrm{d}}$ into a safe controller $\mathbf{k}$.

One advantage of CBFs is their flexibility to account for multiple safety constraints: the CBF-based quadratic programming (CBF-QP) framework above enables a simple integration of multiple CBFs, addressing conjunctive (AND) combination between them. In this case, we note:

$$\begin{aligned} \mathbf{x}\in\textstyle\bigcap_{k=1}^p\mathcal{C}_k \iff \mathbf{x}\in\mathcal{C}_1 \text{ AND } \mathbf{x}\in\mathcal{C}_2 \ \ldots \ \text{AND } \mathbf{x}\in\mathcal{C}_p, \\ \textstyle\bigcap_{k=1}^p\mathcal{C}_k = \big\{x \in \mathbb{R}^n \mid \min\{h_k(\mathbf{x})\}_{k=1}^p \geq 0\big\}, \end{aligned} \tag{5}$$

suggesting the formulation renders the intersection of the sets forward invariant, assuming that the CBFs are compatible and the optimization is feasible for all $\mathbf{x}$ in the set.

This paper investigates logical combinations of constraints beyond AND. Despite the simplicity in dealing with AND combinations, other types like disjunctive (OR) and more complex logical combinations are difficult to handle. For example, OR logic encodes the union of safety constraints:

$$\begin{aligned} \mathbf{x}\in\textstyle\bigcup_{k=1}^p\mathcal{C}_k \iff \mathbf{x}\in\mathcal{C}_1 \text{ OR } \mathbf{x}\in\mathcal{C}_2 \ \ldots \ \text{OR } \mathbf{x}\in\mathcal{C}_p, \\ \textstyle\bigcup_{k=1}^p\mathcal{C}_k = \big\{x \in \mathbb{R}^n \mid \max\{h_k(\mathbf{x})\}_{k=1}^p \geq 0\big\}. \end{aligned} \tag{6}$$

While the AND and OR combinations of safety constraints can be described using the $\min$ and $\max$ functions, these lead to nonsmooth barrier functions [11] and potentially discontinuous controllers if used directly in optimization.

B. Matrix control barrier functions

Matrix control barrier functions (MCBFs) can characterize nonsmooth safe sets from logical compositions between CBFs. Consider the safety constraint defined by a continuously differentiable matrix-valued function $\mathbf{H} : \mathbb{R}^n \to \mathbb{S}^p$:

$$\mathcal{C} = \big\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{H}(\mathbf{x}) \succeq 0\big\}. \tag{7}$$

In contrast to sets defined by scalar-valued functions, the set above can potentially be nonsmooth. Analogous to CBFs, the previous work [21] provides the definition for MCBFs.

Definition 2. *(Matrix CBF, [21]):* A continuously differentiable function $\mathbf{H} : \mathbb{R}^n \to \mathbb{S}^p$ is a *matrix control barrier function* (MCBF) for (1) if there exists $\alpha \in \mathcal{K}^{\mathrm{e}}$ such that, for each $\mathbf{x}$ in the set $\mathcal{C}$ in (7), there exists $\mathbf{u} \in \mathbb{R}^m$ satisfying:

$$\dot{\mathbf{H}}(\mathbf{x}, \mathbf{u}) \triangleq \mathbf{L}_{\mathbf{f}}\mathbf{H}(\mathbf{x}) + \sum_{i=1}^{m} \mathbf{L}_{\mathbf{g}_i}\mathbf{H}(\mathbf{x})\mathbf{u}_i \succ -\boldsymbol{\alpha}(\mathbf{H}(\mathbf{x})), \tag{8}$$

where the matrix function $\boldsymbol{\alpha} : \mathbb{S}^p \to \mathbb{S}^p$ applies α on the eigenvalues of $\mathbf{H}$ while keeping the eigenspaces the same.

We can use MCBFs to enforce safety by posing an optimization problem like (4). In this case, the constraint (8) (with a positive semidefinite inequality, $\succeq$) renders the optimization a semidefinite programming (CBF-SDP).

By enforcing $\mathbf{H}$ to be positive semidefinite, we effectively ensure that all of its eigenvalues are positive at all times. We consider the eigenvalues in ascending order:

$$\lambda_1(\mathbf{x}) \leq \cdots \leq \lambda_p(\mathbf{x}).$$

For diagonal matrices, the eigenvalues are the diagonal entries. Thus, when $\mathbf{H}(\mathbf{x}) = \operatorname{diag}(\{h_k(\mathbf{x})\}_{k=1}^p)$ is the diagonalization of p CBFs, the CBF-SDP makes all $\{h_k(\mathbf{x})\}_{k=1}^p$ remain nonnegative as in (5). In this formulation, the constraint derived from the MCBF is equivalent to the inequalities in (4), addressing the AND composition between CBFs.

[1]For a positive integer p, we denote the set of consecutive numbers as $[p] = \{1, 2, \ldots, p\}$. The set of symmetric matrices in $\mathbb{R}^{p\times p}$ is denoted by $\mathbb{S}^p$, and $\mathbf{I}_p$ is the $p \times p$ identity matrix. For a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with a vector field $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$, we define the Lie derivative $L_{\mathbf{f}}h(\mathbf{x}) = \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x})$. For a continuously differentiable matrix-valued function $\mathbf{H} : \mathbb{R}^n \to \mathbb{S}^p$, the Lie derivative $\mathbf{L}_{\mathbf{f}}\mathbf{H}$ is defined element-wise: $L_{\mathbf{f}}H_{ij}(\mathbf{x}) = \frac{\partial H_{ij}}{\partial \mathbf{x}}(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x})$ where H_{ij} is the (i, j)-th entry of matrix $\mathbf{H}$. Function $\alpha : (-b, a) \to \mathbb{R},\ a, b > 0$ is of extended class-$\mathcal{K}$ ($\alpha \in \mathcal{K}^{\mathrm{e}}$) if it is continuous, strictly increasing, and $\alpha(0) = 0$.

[2]Generally, the safety problem is addressed by finding a CBF h that defines a safe set $\mathcal{C}$ that is a subset of the actual safety constraint.

The work [21] also addresses another type of safe sets:

$$\mathcal{C} = \big\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{H}(\mathbf{x}) \not\succ 0\big\} = \big\{\mathbf{x} \in \mathbb{R}^n \mid \lambda_p(\mathbf{x}) \geq 0\big\} \quad (9)$$

and proposes the following framework.

Definition 3. *(Indefinite-MCBF, [21]):* A continuously differentiable function $\mathbf{H} : \mathbb{R}^n \to \mathbb{S}^p$ is an *indefinite matrix control barrier function* (Indefinite-MCBF) for (1) if there exists $\alpha \in \mathcal{K}^{\mathrm{e}}$ and $c_\perp \geq 0$ such that, for each $\mathbf{x}$ in the set $\mathcal{C}$ in (9), there exists $\mathbf{u} \in \mathbb{R}^m$ satisfying:

$$\dot{\mathbf{H}}(\mathbf{x}, \mathbf{u}) \succ -\alpha\big(\lambda_p(\mathbf{x})\big)\mathbf{I}_p - c_\perp\big(\lambda_p(\mathbf{x})\mathbf{I}_p - \mathbf{H}(\mathbf{x})\big). \quad (10)$$

While the motivation of indefinite-MCBF was to address safe sets that are negations of semidefinite constraints, enabling applications like obstacle avoidance, the framework can also address OR combinations between scalar CBFs. When a diagonal matrix $\mathbf{H}(\mathbf{x}) = \mathrm{diag}(\{h_k(\mathbf{x})\}_{k=1}^p)$ is constructed, the Indefinite-MCBF enforces positivity of the maximal CBF $h_{\max}(\mathbf{x}) = \lambda_p(\mathbf{x})$. Furthermore, when the condition in (10) is diagonal, the corresponding CBF-SDP can be equivalently reformulated into a CBF-QP.

In this paper, we build on the observations on how we can address OR combinations through sorting CBFs, and we generalize the method for addressing *p-choose-r* constraints that, to the best of our knowledge, have not yet been studied.

III. Combinatorial Control Barrier Functions

Next, we address combinatorial safety constraints. Provided p constraints, we enforce that at least r constraints are satisfied at all times (where $r \leq p$), regardless of which constraints. We call this a p-choose-r safety requirement (with a slight abuse of the more precise term p-choose-at-least-r). Such compositional structures naturally arise, for example, when defining safe sets through unions and intersections (OR/AND logic, cf. Example 1) or in fault-tolerant and multi-agent settings where only a subset of components must satisfy a safety condition (Example 2). In these cases, simply neglecting the temporarily inactive constraints can cause control discontinuities, motivating our continuous solution below. Our proposed CBF formulation guarantees the satisfaction of p-choose-r conditions while avoiding the combinatorial blow-up in the number of combinations.

Consider p safety constraints in the form $h_k(\mathbf{x}) \geq 0$, given by functions $\{h_k\}_{k=1}^p$. To satisfy at least r out of these p constraints, $h_k(\mathbf{x}) \geq 0$ must hold for the r-th largest h_k value, which is also the $(p-r+1)$-th smallest h_k, denoted by:

$$h(\mathbf{x}) = \overset{r}{\max}\{h_k(\mathbf{x})\}_{k=1}^p = \overset{p-r+1}{\min}\{h_k(\mathbf{x})\}_{k=1}^p. \quad (11)$$

Then, the safe set $\mathcal{C}$ is defined as in (2). Note that the special cases p-choose-p and p-choose-1 are equivalent to the AND and OR combination of safety constraints in (5)-(6), that is, $\max^p$ and $\max^1$ simplify to $\min$ and $\max$, respectively.

Example 1. *(p-choose-r Constraints):* Figure 2 exemplifies safe sets in two dimensions obtained from p-choose-r compositions. The corner in Fig. 2(a) is a 2-choose-1 constraint:

$$h(\mathbf{x}) = \overset{1}{\max}\{h_k(\mathbf{x})\}_{k=1}^2 = \max\{h_1(\mathbf{x}), h_2(\mathbf{x})\}, \quad (12)$$

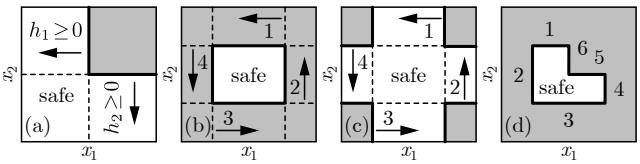

Fig. 2. Examples of safe sets defined by the logic: (a) 2-choose-1, (b) 4-choose-4, (c) 4-choose-3, and (d) 2-choose-2 of 4-choose-4 and 2-choose-1.

encoding that at least one of the two constraints must be satisfied, hence the safe set is the union of two half spaces. The rectangle in Fig. 2(b) implies a 4-choose-4 constraint:

$$h(\mathbf{x}) = \overset{4}{\max}\{h_k(\mathbf{x})\}_{k=1}^4 = \min\{h_1(\mathbf{x}), h_2(\mathbf{x}), h_3(\mathbf{x}), h_4(\mathbf{x})\}, \quad (13)$$

respecting all four boundaries and creating the intersection of four half spaces. Meanwhile, a 4-choose-3 composition:

$$h(\mathbf{x}) = \overset{3}{\max}\{h_k(\mathbf{x})\}_{k=1}^4, \quad (14)$$

captures the cross-shaped region in Fig. 2(c) where at least three out of four constraints hold. •

Inspired by the previous result, we establish the concept of combinatorial CBFs to address p-choose-r safety constraints.

Definition 4. *(Combinatorial CBF):* A function $h : \mathbb{R}^n \to \mathbb{R}$ constructed from sorting CBFs $\{h_k\}_{k=1}^p$ as in (11) is a *combinatorial control barrier function* (p-choose-r CBF) for (1) if there exists $\alpha \in \mathcal{K}^{\mathrm{e}}$ such that, for each $\mathbf{x}$ in the set $\mathcal{C}$ in (2), there exists $\mathbf{u} \in \mathbb{R}^m$ satisfying:

$$\dot{h}_k(\mathbf{x}, \mathbf{u}) > -\alpha\big(h(\mathbf{x}) + |h_k(\mathbf{x}) - h(\mathbf{x})|\big), \quad \forall k \in [p]. \quad (15)$$

The intuition behind the p-choose-r CBF h is that we always ensure that the r-th largest h_k value, which we refer to as the *pivot*, remains nonnegative. While the absolute value term in (15) allows h_k values smaller than h to become negative, it also guarantees that the r-th largest h_k, and thus at least r out of the p functions $\{h_k\}_{k=1}^p$, remain nonnegative.

A p-choose-r CBF enables the following safety result.

Theorem 1. *(Combinatorial Safety): Consider the system* (1)*. Let h be constructed from a combination of primitive CBFs $\{h_k\}_{k=1}^p$ as in* (11)*. If h is a p-choose-r CBF for* (1)*, then the set $\mathcal{C}$ in* (2) *is control invariant (safe).*

In addition, any state feedback controller $\mathbf{k} : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{u} = \mathbf{k}(\mathbf{x})$, that is continuous and satisfies:

$$\dot{h}_k(\mathbf{x}, \mathbf{k}(\mathbf{x})) \geq -\alpha\big(h(\mathbf{x}) + |h_k(\mathbf{x}) - h(\mathbf{x})|\big), \ \forall k \in [p] \quad (16)$$

on a neighborhood $\mathcal{D} \supset \mathcal{C}$, renders the set $\mathcal{C}$ forward invariant and ensures the bounds:

$$\frac{d}{dt}\big(\overset{j}{\max}\{h_k(\mathbf{x}(t)\}_{k=1}^p\big) \geq -\alpha\big(\overset{j}{\max}\{h_k(\mathbf{x}(t)\}_{k=1}^p\big) \quad (17)$$

for all $j \geq r$ at almost every time $t \geq 0$. Consequently, for any initial condition $\mathbf{x}_0 \in \mathcal{C}$, there exist at least r indices $k \in [p]$ at almost every time $t \geq 0$ such that $h_k(\mathbf{x}(t)) \geq 0$.

In particular, the following CBF-QP:

$$\mathbf{k}(\mathbf{x}) = \underset{\mathbf{u} \in \mathbb{R}^m}{\mathrm{argmin}} \quad \|\mathbf{u} - \mathbf{k}_{\mathrm{d}}(\mathbf{x})\|^2 \quad (18)$$

$$\text{s.t.} \quad \dot{h}_k(\mathbf{x}, \mathbf{u}) \geq -\alpha\big(h(\mathbf{x}) + |h_k(\mathbf{x}) - h(\mathbf{x})|\big), \ \forall k \in [p]$$

is a continuous controller satisfying (16).

Proof. Because each $\max^j\{h_k(\mathbf{x})\}_{k=1}^p$ is nonsmooth, the proof relies on nonsmooth barrier function theory. In particular, let $\mathcal{K}_j(\mathbf{x}) = \{k \in [p] \mid h_k(\mathbf{x}) = \max^j\{h_k(\mathbf{x})\}_{k=1}^p\}$ be the set of indices of CBFs with the same value as the function of interest. Then we have the subgradient [22]:

$$\partial\big(\max^j\{h_k(\mathbf{x})\}_{k=1}^p\big) = \mathrm{co} \bigcup_{k\in\mathcal{K}_j(\mathbf{x})} \left\{\frac{\partial h_k}{\partial \mathbf{x}}(\mathbf{x})\right\}$$

$$\implies \frac{d}{dt}\big(\max^j\{h_k(\mathbf{x}(t)\}_{k=1}^p\big) \in \mathrm{co} \bigcup_{k\in\mathcal{K}_j(\mathbf{x})} \{\dot{h}_k(\mathbf{x}(t), \mathbf{u}(t))\}.$$

The implication above follows from the nonsmooth chain rule [22, Thm. 2.3.10] (see also, [11]). Then we note the inequalities in (16) enforce for every $k \in \mathcal{K}_j(\mathbf{x})$:

$$\dot{h}_k(\mathbf{x}(t), \mathbf{u}(t)) \geq -\alpha(h_k(\mathbf{x}(t)))$$

when $j \geq r$, from the definition of the sorting operator $\max^j$. Hence, the bounds given in (17) follow.

Furthermore, when considering the case $j = r$, we derive:

$$\langle \xi,\ \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}(\mathbf{x})\rangle \geq -\alpha(\max^r\{h_k(\mathbf{x})\}_{k=1}^p),$$

for all $\xi \in \partial\big(\max^r\{h_k(\mathbf{x})\}_{k=1}^p\big)$ on the neighborhood $\mathcal{D}$. The function $\max^r\{h_k(\mathbf{x})\}_{k=1}^p$ is therefore a nonsmooth barrier function [11, Prop. 2] for the closed-loop system, verifying the statement of forward invariance.

To prove control invariance, we show that the CBF-QP is a valid safeguarding controller. The definition of the p-choose-r CBFs using strict inequality in (15) ensures that the CBF-QP satisfies Slater's condition at every state $\mathbf{x}$ in a neighborhood $\mathcal{D} \supset \mathcal{C}$ of the safe set, so it is well-defined and continuous there [23]. In addition, the CBF-QP enforces (16) from its constraint, so we may deduce safety. □

Theorem 1 provides the CBF-QP for p-choose-r CBFs. An important feature of this construction is that the number of inequalities in the optimization remains p regardless of r. This avoids the combinatorial blow-up in the numbers of constraint combinations, yielding a more practical complexity that scales linearly with p. This is achieved by sorting the h_k values in (11) to obtain h and then using h as *pivot* in (16) to enforce constraints only for the r largest h_k. The bound (17) ensures forward invariance with the ordered CBFs rather than the individual primitive CBFs, allowing the active subset of r constraints to change dynamically over time. While it may appear natural to apply the CBF condition directly to the r-th largest function (i.e., use $\dot{h} \geq -\alpha(h)$), such an approach involves the derivative of a nonsmooth function, leading to a discontinuous CBF-QP, unlike our proposed framework. Moreover, notice that p-choose-r CBFs define exact safe sets, avoiding conservatism introduced by smooth approximations of h using softmin or softmax operators as done in [13]–[15].

Similar to the case of multiple CBFs in (4), the feasibility of constraint (16) is not automatically guaranteed for CBFs h_k, but it requires h to be a p-choose-r CBF. For instance, in the AND setting (p-choose-p), the existence of a safe input $\mathbf{u}$ is only guaranteed individually for each CBF h_k, while joint feasibility needs to be verified, i.e., h must be a p-choose-p CBF. This remains an active area of research in the CBF literature. In the p-choose-r setting, the situation is subtler: condition (15) must hold even when some $h_k(\mathbf{x}) < 0$, i.e., outside the associated set. There, the absolute value term helps relax this requirement. In principle, the absolute value can be replaced by any positive definite function to help address feasibility and compatibility issues, but this choice recovers the standard AND case when $h(\mathbf{x}) = \min\{h_k(\mathbf{x})\}_{k=1}^p$ and yields the bounds in (17). A full characterization of feasibility is a part of our future work.

Remark 1 (Task Assignment in Multi-Agent Systems)**.** A key motivation for p-choose-r CBFs arises in multi-agent systems. In many settings, safety constraints apply to each agent independently, but the mission requirements involve only subsets of the agents. For instance, in a surveillance task, the collective objective may be to ensure that a sensitive region is not left unattended, and it is sufficient that at least r agents remain within the region. Indeed, while mission planning typically handles such task allocation decisions, embedding this logic into combinatorial CBFs enhances safety assurance at the faster control layer, complementing the slower and more uncertainty-prone planning layer. In addition, unlike the single-agent case, feasibility and compatibility are less problematic here since the agents' dynamics and control inputs are decoupled, allowing the constraints to be satisfied in parallel. The following example uses a p-choose-r CBF in a multi-agent task assignment scenario. •

Example 2. *(Surveillance Task):* Consider a circular region shown in Fig. 1 and a system of $p = 3$ robots with dynamics:

$$\dot{\mathbf{x}}_k = \mathbf{u}_k,\ k \in [p].$$

We can use a p-choose-r CBF to assign at least $r = 2$ robots out of $p = 3$ to remain inside the circle at all times. Define the state $\mathbf{x} = [\mathbf{x}_1^\top\ \mathbf{x}_2^\top\ \ldots\ \mathbf{x}_p^\top]^\top$ and the following functions:

$$h_k(\mathbf{x}) = R^2 - \|c - \mathbf{x}_k\|^2,\ k \in [p],$$
$$h(\mathbf{x}) = \max^r\{h_k(\mathbf{x})\}_{k=1}^p.$$

where R and c are the radius and the center of the region. Each function $h_k(\mathbf{x}) \geq 0$ indicates that the robot k lies inside the circle, so $h(\mathbf{x})$ is a p-choose-r CBF which is nonnegative if and only if at least r robots are inside the circle. •

IV. Nested Logic for Safety Constraints

We now extend our framework to handle nested combinatorial safety constraints, where multiple combinatorial CBFs are composed hierarchically. This generalization builds on the same sorting and pivoting arguments used for p-choose-r CBFs, but applied recursively to follow the nested structure. The resulting formulation encodes the overall safety requirement through a single pivot safety function h, which plays the role of the effective barrier function for the nested logic.

For simplicity, we first focus on two-level nested logic. Consider a p^2-choose-r^2 combination of multiple p_i^1-choose-r_i^1 combinations among some i-th subset of *primitive* CBFs

$\{h_k\}_{k=1}^{p}$. Note carefully that the superscripts on p's and r's indicate the nested level of the logical combinations, rather than exponentiation. We define the safe set $\mathcal{C}$ as in (2) with:

$$\begin{aligned} h_i^1(\mathbf{x}) &= \overset{r_i^1}{\max}\{h_k(\mathbf{x})\}_{k\in\mathcal{I}_i}, \quad i \in [p^2], \\ h(\mathbf{x}) &= \overset{r^2}{\max}\{h_k^1(\mathbf{x})\}_{k=1}^{p^2}, \end{aligned} \tag{19}$$

where $\mathcal{I}_i$ denote the indices of the p_i^1 constraints that are combined through the p_i^1-choose-r_i^1 logic. Here, we reveal that the nested logic is a logical combination of combinatorial CBFs $\{h_k^1\}_{k=1}^{p^2}$. We provide the following concrete example.

Example 3. *(Nested Constraints):* Consider the L-shaped region in Fig. 2(d) that is the intersection of the rectangle in Fig. 2(b) and the corner in Fig. 2(a). The L-shape is described by a 2-choose-2 (AND) combination of a 4-choose-4 constraint (rectangle) and a 2-choose-1 (corner):

$$\begin{aligned} h_1^1(\mathbf{x}) &= \overset{4}{\max}\{h_k(\mathbf{x})\}_{k=1}^{4},\ h_2^1(\mathbf{x}) = \overset{1}{\max}\{h_k(\mathbf{x})\}_{k=5}^{6}, \\ h(\mathbf{x}) &= \overset{2}{\max}\{h_k^1(\mathbf{x})\}_{k=1}^{2}, \end{aligned} \tag{20}$$

see (13) and (12) in Example 1. •

Naively, one would deal with the two-level nested logic with inequalities making h a p^2-choose-r^2 CBF of $\{h_k^1\}_{k=1}^{p^2}$:

$$\dot{h}_k^1(\mathbf{x},\mathbf{u}) > -\alpha\big(h(\mathbf{x}) + |h_k^1(\mathbf{x}) - h(\mathbf{x})|\big), \quad \forall k \in [p^2].$$

However, the main concern here is the nonsmoothness associated with h_k^1. We can avoid this issue by applying the barrier condition in (16) directly on the primitive functions, like for one-level logical combinations. Intuitively, while h_k^1 is nonsmooth, its derivative $\dot{h}_k^1$ remains a convex combination of some primitive $\dot{h}_k$'s. Therefore, by enforcing (16) on all primitives, we implicitly enforce the desired inequalities above, rendering h a p^2-choose-r^2 CBF of $\{h_k^1\}_{k=1}^{p^2}$. In other words, the one-level barrier conditions in (16) extend directly to the two-level case, with the only change being how h is defined. This observation generalizes by induction to any number of nested logical levels.

Consider the case when p primitive CBFs $\{h_k\}_{k=1}^{p}$ are combined into a single constraint through M levels of safety specifications with index $\ell \in [M]$. At each level, the constraints are combined via p_i^ℓ-choose-r_i^ℓ logic to create p^ℓ new safety constraints, described by functions $\{h_i^\ell\}_{i=1}^{p^\ell}$:

$$\begin{aligned} h_i^0(\mathbf{x}) &= h_i(\mathbf{x}), \quad i \in [p], \\ h_i^\ell(\mathbf{x}) &= \overset{r_i^\ell}{\max}\{h_k^{\ell-1}(\mathbf{x})\}_{k\in I_i^\ell}, \quad i \in [p^\ell], \quad \ell \in [M], \\ h(\mathbf{x}) &= h_1^M(\mathbf{x}), \end{aligned} \tag{21}$$

where I_i^ℓ denote the indices of constraints that are combined to obtain constraint i at level ℓ, with $I_i^\ell \subseteq [p^{\ell-1}]$, $|I_i^\ell| = p_i^\ell \le p^{\ell-1}$ for all $i \in [p^\ell]$, $\ell \in [M]$. With the function h obtained at the last level, the safe set $\mathcal{C}$ is defined in (2). We summarize our findings with the following theorem.

Theorem 2. *(Nested Logic Safety): Consider the system* (1)*. Let h be constructed from a multi-level nested logical combination of primitive CBFs $\{h_k\}_{k=1}^{p}$ as in* (21)*. If h is a combinatorial CBF for* (1)*, then the set $\mathcal{C}$ in* (2) *is safe.*

In addition, any state feedback controller $\mathbf{k} : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{u} = \mathbf{k}(\mathbf{x})$, that is continuous and satisfies (16) *on a neighborhood $\mathcal{D} \supset \mathcal{C}$ renders the set $\mathcal{C}$ forward invariant. The CBF-QP* (18) *is one such controller.* ■

We omit the proof for brevity. It follows the same structure as Theorem 1 and proceeds by induction on the nesting levels. The key idea is that the subgradient set can eventually be expressed in terms the gradients of the primitive CBFs. Hence, nested logic introduces no additional complexity beyond redefining the pivot function h.

We emphasize that, despite encoding nested logical safety requirements, our formulation requires only the original p primitive constraints. This eliminates the combinatorial blow-up in the number of combinations and significantly reduces the problem size compared to a naive implementation.

Remark 2 (Logical Composition of MCBFs)**.** We have described logical compositions starting from primitive scalar CBFs. The same ideas naturally extend to MCBFs, which include scalar CBFs as a special case. In this setting, the role of sorting primitive functions is played by sorting the eigenvalues of the primitive MCBFs, $\{\mathbf{H}_k\}_{k=1}^{p}$. In this framework, each MCBF can be interpreted as the first level in a nested CBF: a MCBF corresponds to enforcing a minimum (AND) over its eigenvalues, while an indefinite-MCBF corresponds to enforcing a maximum (OR). A detailed analysis and full technical treatment of this setting are left for future work. •

V. Simulation

In this section, we illustrate our results using a multi-agent patrolling problem. Consider two separated L-shaped regions, $L_1, L_2 \subset \mathbb{R}^2$, as shown in Fig. 3, that are monitored by $N = 11$ agents. Each agent follows the dynamics:

$$\dot{\mathbf{x}}_j = \mathbf{u}_j,\ j \in [N] \tag{22}$$

and is assigned a desired patrolling controller:

$$\mathbf{k}_{\mathrm{d},j}(\mathbf{x},t) = \big[\kappa_j\big(x_{\mathrm{d},j}(t) - x_j\big) + \dot{x}_{\mathrm{d},j}(t),\ 0\big]^\top, \tag{23}$$

where $x_{\mathrm{d},j}(t) = A_j \sin(\omega_j t)$ is the desired patrolling trajectory, with $\kappa_j, A_j, \omega_j > 0$ denoting the proportional gain, motion amplitude, and patrolling frequency, respectively. Here, heterogeneity among the agents is abstracted by varying their patrolling frequency, reflecting their differences in capability.

To ensure robust monitoring against errors in sensor measurements, agent failures, etc., we require at least four agents to remain within each region L_1 and L_2 at all times. To enforce such requirement using Boolean compositions, we would need to consider a total of $\sum_{j=4}^{11}\binom{11}{j}\sum_{i=1}^{2}\binom{2}{i} = 1816 \times 3 = 5448$ combinations, which would not be computationally practical. Therefore, we define the following functions based on the proposed method:

$$\begin{aligned} h_{L_1}(\mathbf{x}_j) &= \min\Big\{\min\{h_k(\mathbf{x}_j)\}_{k=1}^{4},\ \max\{h_k(\mathbf{x}_j)\}_{k=5}^{6}\Big\}, \\ h_{L_2}(\mathbf{x}_j) &= \min\Big\{\min\{h_k(\mathbf{x}_j)\}_{k=7}^{10},\ \max\{h_k(\mathbf{x}_j)\}_{k=11}^{12}\Big\}, \\ h(\mathbf{x}) &= \min\Big\{\overset{4}{\max}\{h_{L_1}(\mathbf{x}_j)\}_{j=1}^{11},\ \overset{4}{\max}\{h_{L_2}(\mathbf{x}_j)\}_{j=1}^{11}\Big\}. \end{aligned} \tag{24}$$

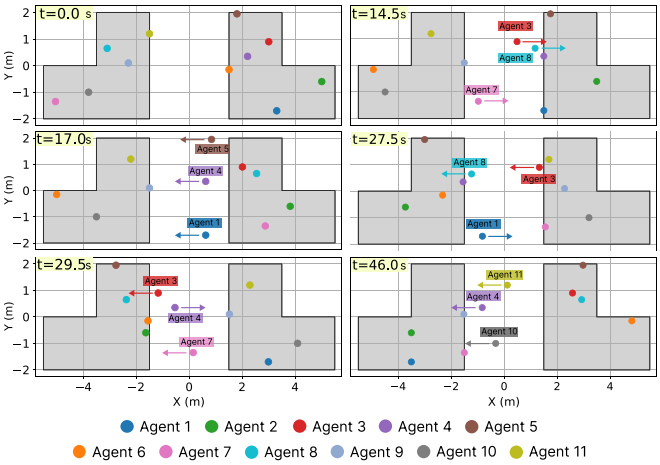

Fig. 3. Simulation of a multi-agent patrolling problem where each L-shaped region must be monitored by at least four agents at all times. Using the proposed CBF-QP (18), this combinatorial safety constraint is ensured in a computationally tractable manner. The code and animation are available at `https://github.com/joonlee16/combinatorial_cbf`.

Here, h_{L_1} and h_{L_2} characterize the two L-shaped regions using (20) from Example 3 (a 2-choose-2 combination of 4-choose-4 and 2-choose-1). If $h_{L_1}(\mathbf{x}_j) \geq 0$ or $h_{L_2}(\mathbf{x}_j) \geq 0$, then agent j's position resides within the L_1 or L_2 region, respectively. Using 11 instances of h_{L_1} and h_{L_2} (one per agent), we construct the combinatorial CBF h as given in (24) (a 2-choose-2 combination of 11-choose-4). The condition $h(\mathbf{x}) \geq 0$ ensures that at least four agents are in each region simultaneously. By Theorem 2, we can enforce safety through the CBF-QP (18) with only $p = 132$ constraints.

We simulated the motions of the agents using the dynamics (22), the CBF-QP (18), and the p-choose-r CBF (24). Snapshots for $t \in [0, 50]$ are shown in Fig. 3, with colored arrows indicating agents moving into and out of the L-shaped regions. Because each region must always be covered by at least four agents, at most three agents can be outside both regions at any time. As shown in the figure, this constraint is satisfied throughout the simulation.

VI. Conclusion

We have developed a framework for the logical composition of CBFs and MCBFs that extends beyond simple AND formulations. The proposed combinatorial CBFs not only address p-choose-r safety constraints but also their nested logical combinations. Similar to the standard AND case, our formulation enforces safety through multiple inequalities, thereby avoiding the nonsmoothness issues associated with logical operations. The key idea is to correctly identify the safety function via a sorting-based argument and utilize it as a pivot in the barrier conditions. Our future work will explore how to handle the feasibility issue arising when CBF conditions must hold outside their original safe sets and the compatibility issue arising in the intersection of their sets.

References

[1] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *European Control Conference*, (Naples, Italy), pp. 3420–3431, June 2019.

[2] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[3] M. Rauscher, M. Kimmel, and S. Hirche, "Constrained robot control using control barrier functions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (Daejeon, South Korea), pp. 279–285, Oct. 2016.

[4] X. Xu, "Constrained control of input–output linearizable systems using control sharing barrier functions," *Automatica*, vol. 87, pp. 195–201, 2018.

[5] L. Wang, A. D. Ames, and M. Egerstedt, "Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots," in *IEEE Conference on Decision and Control*, (Las Vegas, NV), pp. 2659–2664, Dec. 2016.

[6] X. Tan and D. V. Dimarogonas, "Compatibility checking of multiple control barrier functions for input constrained systems," in *IEEE Conference on Decision and Control*, (Cancún, Mexico), pp. 939–944, Dec. 2022.

[7] J. Breeden and D. Panagou, "Compositions of multiple control barrier functions under input constraints," in *American Control Conference*, (San Diego, CA), pp. 3688–3695, May 2023.

[8] H. Lee, P. Rousseas, and D. Panagou, "Constraint selection in optimization-based controllers," *arXiv preprint*, no. 2505.05502, 2025.

[9] M. H. Cohen, E. Lavretsky, and A. D. Ames, "Compatibility of multiple control barrier functions for constrained nonlinear systems," in *IEEE Conference on Decision and Control*, (Rio de Janeiro, Brazil), Dec. 2025. To appear.

[10] K. H. Kim, M. Diagne, and M. Krstić, "Constant-sum high-order barrier functions for safety between parallel boundaries," *IEEE Control Systems Letters*, vol. 9, pp. 1447–1452, 2025.

[11] P. Glotfelter, J. Cortés, and M. Egerstedt, "Nonsmooth barrier functions with applications to multi-robot systems," *IEEE Control Systems Letters*, vol. 1, no. 2, pp. 310–315, 2017.

[12] Y. Yang, C. Manzie, and Y. Pu, "A control barrier function composition approach for multiagent systems in marine applications," *IEEE/ASME Transactions on Mechatronics*, pp. 1–12, 2025.

[13] T. G. Molnar and A. D. Ames, "Composing control barrier functions for complex safety specifications," *IEEE Control Systems Letters*, vol. 7, pp. 3615–3620, 2023.

[14] T. G. Molnar, "Navigating polytopes with safety: A control barrier function approach," in *IEEE Conference on Control Technology and Applications*, (San Diego, CA), pp. 179–184, Aug. 2025.

[15] M. Black and D. Panagou, "Adaptation for validation of consolidated control barrier functions," in *IEEE Conference on Decision and Control*, (Marina Bay Sands, Singapore), pp. 751–757, Dec. 2023.

[16] M. Harms, M. Jacquet, and K. Alexis, "Safe quadrotor navigation using composite control barrier functions," in *IEEE International Conference on Robotics and Automation*, (Atlanta, GA), May 2025.

[17] C. Wang, X. Wang, Y. Dong, L. Song, and X. Guan, "Multi-constraint safe reinforcement learning via closed-form solution for log-sum-exp approximation of control barrier functions," in *Proceedings of Machine Learning Research*, vol. 283, pp. 698–710, June 2025.

[18] R. Lin and M. Egerstedt, ""Hierarchy of needs" for robots: Control synthesis for compositions of hierarchical, complex objectives," in *IEEE International Conference on Robotics and Automation*, (Atlanta, GA), pp. 7682–7688, May 2025.

[19] L. Lindemann and D. V. Dimarogonas, "Control barrier functions for signal temporal logic tasks," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 96–101, 2019.

[20] L. Lindemann and D. V. Dimarogonas, "Control barrier functions for multi-agent systems under conflicting local signal temporal logic tasks," *IEEE Control Systems Letters*, vol. 3, no. 3, pp. 757–762, 2019.

[21] P. Ong, Y. Xu, R. M. Bena, F. Jabbari, and A. D. Ames, "Matrix control barrier functions," *IEEE Transactions on Automatic Control*, 2025. Submitted.

[22] F. H. Clarke, *Optimization and Nonsmooth Analysis*. Canadian Mathematical Society Series of Monographs and Advanced Texts, New York: Wiley, 1983.

[23] P. Mestres, A. Allibhoy, and J. Cortés, "Regularity properties of optimization-based controllers," *European Journal of Control*, vol. 81, p. 101098, 2025.