Received: 7 March 2023 | Revised: 19 September 2023 | Accepted: 3 December 2023

DOI: 10.1002/rnc.7141

RESEARCH ARTICLE

WILEY

Feasibility-guaranteed safety-critical control with high-order control barrier function method

Sihua Zhang[1] | **Di-Hua Zhai**[1,2] | **Yuhan Xiong**[1] | **Yuanqing Xia**[1]

[1]School of Automation, Beijing Institute of Technology, Beijing, China

[2]Yangtze Delta Region Academy of Beijing Institute of Technology, Jiaxing, China

Correspondence
Di-Hua Zhai, School of Automation, Beijing Institute of Technology, Beijing, China.
Email: zhaidih@bit.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 62173035, 61803033, 61836001; Xiaomi Young Scholars from Xiaomi Foundation

Abstract

The optimization of control systems under the presence of safety constraints and input constraints frequently involves the decomposition into a sequence of quadratic programs (QPs) facilitated by the utilization of high-order control barrier function (HOCBF). When the safety constraint conflicts with the input constraint, however, it leads to infeasibility within the QPs. In this article, a feasibility-guaranteed QP is proposed to tackle the challenge posed by the conflict between HOCBF constraint and input constraint. Firstly, the classical QP is added with a feasibility constraint which is derived from input constraint and HOCBF constraint, where the parameter of feasibility constraint is updated via a new QP obtained by control sharing property. Then, Type-2 HOCBF is investigated for the system with multiple HOCBF constraints, which effectively confines the system within a single HOCBF at the current time step. Finally, the efficacy of this approach is demonstrated through the application of obstacle avoidance in a 3-DOF robot system.

KEYWORDS

control barrier function, feasibility constraint, quadratic program, safety-critical control

1 | INTRODUCTION

With the growing complexity of systems, guaranteeing safety properties becomes paramount in controller design.[1] Applications such as autonomous driving and industrial robotics underscore the significance of synthesizing controllers that prioritize safety. Among the various approaches, optimization-based techniques like model predictive control (MPC)[2] are an appealing choices for designing safety-critical controller. However, it is challenging to solve the optimization problem in real-time. Recently, the adoption of control barrier functions (CBFs) to ensure forward invariance of safe sets has received increasing attention.[3,4]

Control barrier function (CBF) is first defined for safety constraint with relative degree one in Reference 5, which serves as a mechanism to transform state-based constraints into constraints that are expressed in terms of control inputs. Moreover, some methods are also developed to handle the constraints with high relative degree. A backstepping approach is developed in Reference 6 for relative degree two. An exponential CBF (ECBF) is investigated for constraints with arbitrarily high relative degree using tools from linear control theory in Reference 7. The creation of ECBF is based on pole placement, but the choice of pole location requires care as it depends on the initial condition. In this case, the high order CBF (HOCBF) is developed,[8] which is both more general and simpler to employ when dealing with constraints of high relative degree compared to ECBF.

Most of the existing works uses the framework combining CBF with control Lyapunov function (CLF)[9] through quadratic programming (QP) to enforce safety and stability.[10] In this framework, the time domain is discretized based on the assumption that the control input and state are constant over each time step.[11] However, this approach may encounter QP infeasibility issues. Specifically, when a constant control input is maintained across two adjacent control steps, it can lead to incompatible constraints, rendering the QP infeasible in the subsequent control interval. For instance, if the control objective falls outside the safe set defined by the safety constraint, the conflicting CLF and CBF constraints can render the QP infeasible. For safety-critical systems, the CLF constraints are relaxed as soft constraints, while the CBF constraints remain unchanged as hard constraints on the QP. In this way, QP becomes feasible because both security and stability do not need to be satisfied.

Moreover, the CBF constraint may be incompatible with the control input bound. To address this issue, various approaches such as the penalty method[12] and adaptive CBF[13] have been developed to ensure the feasibility of QP. However, since no closed-form solutions are derived and the penalties are tuned via numerical techniques, those methods are difficult to further study the performance of system in constrained optimal control problems. Reference 14 proposes an optimal-decay form for safety-critical control wherein the decay rate of the CBF is optimized point-wise in time so as to guarantee point-wise feasibility when the state lies inside the safe set. The decay-rate relaxing technique is generalized for MPC with discrete-time high-order control barrier function (DHOCBF) in Reference 15. It presented a method to enhance the feasibility of iterative optimization subject to linearized DHOCBF by relaxing the decay rate in each constraint. However, when the states reach the boundary of safe set, the feasibility of optimization cannot be guaranteed by the decay-rate relaxing method. The backup CBF is proposed as a tractable formulation that guarantees the feasibility of QP via an implicitly defined control invariant set in Reference16. The control invariant set is based on a fixed backup policy and evaluated online by forward integrating the dynamics under the backup policy. However, this method substantially increases the computational complexity of the optimization problem and a good backup policy is very complicated for complex systems. In Reference 17, sufficient conditions are first captured by a CBF constraint and added to the QP to guarantee feasibility. However, how to find the candidate function that satisfies the sufficient conditions is still a problem.

Addressing the feasibility problem in the presence of multiple safety constraints and control bounds is notably challenging, as it necessitates the avoidance of conflicts among all these constraints. Unfortunately, little work has been investigated on handling multiple safety constraints and input constraints simultaneously. Reference 18 investigates a method for handling the multiple CBFs with input constraints, but it requires some model knowledge, including Lipschitz constants and dynamics bounds. A robust multiple CBFs framework is proposed for the passivity-based system with input constraints in Reference 19. In addition, Breeden and Panagou20 guarantees the feasibility of QP subject to multiple CBF constraint and input constraint by providing tools to decouple the design of multiple CBF, so that a CBF can be designed for each constraint function independently of other constraints, and ensure that the set composed from all the CBFs together is a viability domain. However, the methods in References 19 and 20 are applicable exclusively to systems with a relative degree of one.

In this article, a novel safety-critical method is proposed based on HOCBF, aiming to solve the problem that a safety constraint may conflict with input constraint and make QP infeasible. To mitigate this issue, a feasibility constraint is introduced into the QP formulation. This constraint is derived from both a safety constraint and an input constraint, taking the form of a CBF constraint with a relative degree of one. The parameter of feasibility constraint is updated by using a new QP, which leverages the control sharing property to ensure compatibility among all constraints. Moreover, if the system has multiple safety constraints, a Type-2 HOCBF is designed to transform multiple safety constraints into a single safety constraint, thereby simplifying the process of ensuring QP feasibility. Finally, numerical simulation is performed on a 3-DOF plane robot to verify the method.

1.1 Notations

$\mathbb{R}^n$ denotes the n-dimensional Euclidean space and $\mathbb{R}^{n\times n}$ denotes a space of real matrices with n rows and m columns. $\|\cdot\|$ denotes the 2-norm of the vector. $|\cdot|$ denotes the absolute value. For an n-dimensional vector $x = [x_1, \dots, x_n]$, $x_i (i = 1, 2, \dots, n)$ represents the ith element of x. $L_f b(x)$ denotes the Lie derivative of $b(x)$ along f at x, and $L_f b(x) = \frac{\partial b(x)}{\partial x} f(x)$. $L_f^m b(x)$ denotes the m-order Lie derivative of $b(x)$ along f at x. q, $\dot{q}$ and $\ddot{q}$ denote the joint position, velocity and acceleration of robotic system, respectively.

2 | PRELIMINARIES

Consider the affine control system

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

where $x \in \mathbb{R}^n, f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times q}$ are locally Lipschitz, and $u \in U \subset \mathbb{R}^q$ is the control input. U denotes the input constraint set and satisfies

$$U = \{u \in \mathbb{R}^q : u_{min} \leq u \leq u_{max}\}. \tag{2}$$

The closed set C defined by a continuously differentiable function $b(x) : \mathbb{R}^n \to \mathbb{R}$ is given as follows:

$$C = \{x \in \mathbb{R}^n : b(x) \geq 0\}, \tag{3}$$
$$\text{Int}(C) = \{x \in \mathbb{R}^n : b(x) > 0\}, \tag{4}$$
$$\partial C = \{x \in \mathbb{R}^n : b(x) = 0\}, \tag{5}$$

It is assumed that C is nonempty and has no isolated point. When the set C is forward invariant, the system (1) is safe and the set C is called safe set. The definition of forward invariant is given below.

Definition 1 (Forward Invariant[21]). The set $C \subset \mathbb{R}^n$ is forward invariant for system (1) if its solution starting at any $x(0) \in C$ satisfies $x(t) \in C$ for all $t \geq 0$.

To ensure set invariance, the CBF method is derived. Firstly, some important definitions are introduced.

Definition 2 (Class $\mathcal{K}$ Function[22]). A continuous function $\alpha : [0, a) \to [0, \infty), a > 0$, is a class $\mathcal{K}$ function if it is strictly increasing and $\alpha(0) = 0$.

Definition 3 (Relative degree[22]). For a continuously differentiable function $b(x) : \mathbb{R}^n \to \mathbb{R}$ with respect to system (1), the relative degree is the number of times it needs to be differentiated along its dynamics until the control input explicitly shows in the corresponding derivative.

Suppose that the relative degree of function $b(x)$ is m, and the inequality $b(x) \geq 0$ is used as a constraint with the relative degree of m. If $m = 1$, then the definition of CBF is given as below.

Definition 4 (Control Barrier Function[7]). Given a set C as in (3), $b(x)$ is a CBF for system (1) if there exists a class $\mathcal{K}$ function α such that

$$\sup_{u\in\mathbb{R}^q} [L_f b(x) + L_g b(x)u + \alpha(b(x))] \geq 0, \quad \forall x \in C, \tag{6}$$

where L_f and L_g denote the Lie derivatives along f and g, respectively.

If $m > 1$, then the CBF can not be used to guarantee the forward invariance of set since the control input u is no longer exhibited in (6). So, the HOCBF is proposed. A sequence of functions $\psi_i(x) : \mathbb{R}^n \to \mathbb{R}, i \in \{0, \dots, m\}$ are first defined as

$$\psi_0(x) = b(x),$$
$$\psi_i(x) = \dot{\psi}_{i-1}(x) + \alpha_i(\psi_{i-1}(x)), \quad i \in \{1, \dots, m\}, \tag{7}$$

where $\alpha_i(\cdot)$ denotes $(m - i)^{th}$ order differentiable $\mathcal{K}$-class function. A sequence of sets $C_i, i \in \{1, \dots, m\}$ are then defined in the form of

$$C_i = \{x \in \mathbb{R}^n : \psi_{i-1}(x) \geq 0\}, \quad i \in \{1, \dots, m\}. \tag{8}$$

Then, the definition of high-order control barrier function (HOCBF) is given as below.

Definition 5 (HOCBF[12]). A function $b(x) : \mathbb{R}^n \to \mathbb{R}$ is a High Order Control Barrier Function (HOCBF) of relative degree m for system (1) if there exist $(m - i)^{th}$ order differentiable class $\mathcal{K}$ functions $\alpha_i, i \in \{1, \dots ,$

$m-1\}$, and a class $\mathcal{K}$ functions α_m such that

$$\sup_{u\in\mathbb{R}^q}[L_f^m b(x) + L_g L_f^{m-1} b(x)u + S(b(x)) + \alpha_m(\psi_{m-1}(x))] \geq 0, \tag{9}$$

for all $x \in C_1 \cap \cdots \cap C_m$. L_f^m denotes m-order Lie derivative along f in the system (1). $S(b_i(x))$ denotes the remaining Lie derivative along f with degree less than or equal to $m-1$, that is,

$$S(b_i(x)) = \sum_{i=1}^{m-1} L_f^i(\alpha_{m-i} \circ \psi_{m-i-1})(x). \tag{10}$$

Lemma 1 ([12]). *The set $C_1 \cap \cdots \cap C_m$ is forward invariant for system (1), if $x(0) \in C_1 \cap \cdots \cap C_m$ and $b(x)$ is a HOCBF. The Lipschitz continuous control input u belongs to the set*

$$K_s(x) = \{u \in \mathbb{R}^q : -L_g L_f^{m-1} b(x)u \leq L_f^m b(x) + S(b(x)) + \alpha_m(\psi_{m-1}(x))\}. \tag{11}$$

To find a control policy for system (1), the most existing works form an optimal control problem by combining HOCBF with quadratic costs in control u as below.

$$\begin{aligned} & J(u(t)) = \int_0^T \|u(t) - u_{nom}(t)\|^2 dt \\ & s.t. \\ & L_f^m b(x) + L_g L_f^{m-1} b(x)u + S(b(x)) + \alpha_m(\psi_{m-1}(x)) \geq 0, \\ & u_{min} \leq u \leq u_{max}, \end{aligned} \tag{12}$$

where $\|\cdot\|$ denotes the 2-norm of a vector, u_{nom} is a nominal feedback controller. When $u_{nom} \notin K_{cbf}(x)$, the CBF constraint will minimally modify the nominal controller to ensure safety. The optimal control problem is usually solved point-wise, where the time interval $[0, T]$ is divided into a finite number of intervals $[t_k, t_{k+1})$, $k = 0, 1, 2 \ldots n$. Besides, the constraint is linear in control and the states are fixed at each interval, so that the optimization problem eventually becomes a quadratic programming (QP) as follows:

$$\begin{aligned} & u^* = \arg\min_u \|u - u_{nom}\|^2 \\ & s.t. \\ & L_f^m b(x) + L_g L_f^{m-1} b(x)u + S(b(x)) + \alpha_m(\psi_{m-1}(x)) \geq 0, \\ & u_{min} \leq u \leq u_{max}. \end{aligned} \tag{13}$$

Therefore, the optimal control input is obtained by solving a QP at each interval, updating states through dynamics (1), and then repeating the procedures.

Remark 1. Most existing work on implementing safety-critical control with HOCBF solves the QP (13) in each interval using the above method. This conversion is first applied in Reference 4, which proves the control input from QP is locally Lipschitz continuous and a closed-form expression can be given for control input. It is important to note that this method is computationally efficient, but continuous safety may not be satisfied between time intervals. To address this issue,[23,24] present improved methods by bounding the time derivative of the CBF between time intervals. Given that the above solution is complicated and is out of the main focus of this article, the proposed framework only sets the time intervals to be small to avoid this problem.

3 | FEASIBILITY-GUARANTEED QP

For the system with input constraint, if the safety constraint is incompatible with the input constraint, the QP becomes infeasible. This section presents a feasibility-guaranteed method for the QP and designs the safety-critical controller.

3.1 Feasibility constraint

For system (1), the QP subject to HOCBF and input constraint can be written as (13). The HOCBF condition enforces that the system always satisfies safety constraint $b(x) \geq 0$. Also, the input constraint should always be satisfied. When the safety constraint conflicts with input constraint, the QP (13) is infeasible, making it impossible to obtain control input. To get the control input that satisfies safety constraint and input constraint, the feasibility constraint is investigated.

Definition 6 (Feasibility Constraint[17]). Suppose that the QP (13) is feasible at the current state $x(\bar{t})$, $\bar{t} \in [t_k, t_{k+1})$. For constraint $b_F(x) : \mathbb{R}^n \to \mathbb{R}$, $b_F(x) \geq 0$ is a feasibility constraint if it makes the QP (13) corresponding to the next time interval $[t_{k+1}, t_{k+2})$ feasible.

The feasibility constraint $b_F(x) \geq 0$ should satisfy two requirements to guarantee the feasibility of QP (13) at next time interval. Firstly, it should ensure that the safety constraint and input constraint do not conflict. Second, the feasibility constraint cannot conflict with both safety constraint and input constraint at the same time. How to obtain the CBF constraint will be given later.

The analysis for the feasibility constraint depends on two sets defined by these constraints in terms of u. The first set is shown in (11), where the control input multiplies the vector $-L_gL_f^{m-1}b(x) = [-L_gL_f^{m-1}b(x)_1, \cdots, -L_gL_f^{m-1}b(x)_q]$. The second set is based on (2) by multiplying the vector $-L_gL_f^{m-1}b(x)$, that is,

$$U_c = \{u \in \mathbb{R}^q : u_l(x) \leq -L_gL_f^{m-1}b(x)u \leq u_d(x)\}, \tag{14}$$

$$u_l(x) = \sum_{i=1}^{q} \min(-L_gL_f^{m-1}b(x)_i u_{min,i}, -L_gL_f^{m-1}b(x)_i u_{max,i}), \tag{15}$$

$$u_d(x) = \sum_{i=1}^{q} \max(-L_gL_f^{m-1}b(x)_i u_{min,i}, -L_gL_f^{m-1}b(x)_i u_{max,i}). \tag{16}$$

It is obvious that $U \subset U_c$. The following lemma shows that the relaxation of U has no negative effect on the compatibility of safety constraint and input constraint.

Lemma 2. [17] *If the intersection of the two sets in (11) and (14) is non-empty, then the intersection of the two sets in (11) and (2) is non-empty for all x.*

Therefore, the condition that the intersection of two sets $K_s(x)$ and U_c is non-empty for all x guarantees no-conflict of safety constraint and input constraint. According to the definitions of $K_s(x)$ in (11) and U_c in (14), the condition is ensured if

$$L_f^m b(x) + S(b(x)) + \alpha_m(\psi_{m-1}(x)) \geq u_l(x). \tag{17}$$

(17) can be seen as a feasibility constraint for the QP (13), which is designed to be a CBF $b_F(x)$ to guarantee the satisfactory of constraint (17). The relative degree of $b_F(x)$ is one with respect to dynamics (1).

$$b_F(x) = L_f^m b(x) + S(b(x)) + \alpha_m(\psi_{m-1}(x)) - u_l(x) \geq 0. \tag{18}$$

Then, if $b_F(x_0) \geq 0$, then the following controller $K_F(x)$ can guarantee the free conflict between HOCBF constraint (9) and input constraint.

$$K_F(x) = \{u \in \mathbb{R}^q : L_f b_F(x) + L_g b_F(x)u + k_F\alpha_F(b_F(x)) \geq 0\}, \tag{19}$$

where $\alpha_F(b_F(x))$ is a $\mathcal{K}$-class function and k_F is a positive constant.

By adding the feasibility constraint of $b_F(x)$ into QP (13), the QP is then written as follows:

$$\begin{aligned}
&u^* = \arg\min_u \|u - u_{nom}\|^2 \\
&s.t. \\
&L_f^m b(x) + L_gL_f^{m-1}b(x)u + S(b(x)) + \alpha_m(\psi_{m-1}(x)) \geq 0, \\
&L_f b_F(x) + L_g b_F(x)u + k_F\alpha_F(b_F(x)) \geq 0, \\
&u_{min} \leq u \leq u_{max},
\end{aligned} \tag{20}$$

To guarantee the feasibility of the QP (20), it is need to be satisfied that all constraints in QP (20) are compliant with each other. In this regard, similar to Reference 25, the control sharing property is used.

Definition 7 (Control Sharing Property[25]). For system (1), the CBFs $b_i(x) : \mathbb{R}^n \to \mathbb{R}$ with relative degree r_i are said to have control sharing property, if there exists control input $u \in U$ such that the following condition holds for all $i = 1, \dots, q$:

$$L_f^{r_i} b_i(x) + L_g L_f^{r_i-1} b_i(x)u + S(b_i(x)) + \alpha_{r_i}^i(\psi_{r_i-1}(x)) \geq 0. \tag{21}$$

The feasibility of (20) requires that the feasibility constraint and safety constraint should have control sharing property. According to the derivation process that the condition of control sharing property in,[25] the condition that makes all constraints compliant with each other is summarized in the following theorem.

Theorem 1. *Given the QP (20), when $b_F(x) > 0$, if k_F is obtained by solving the following QP for all $i \in \{k \in \mathbb{R} : L_g b_F(x)_k L_g L_f^{m-1} b(x)_k \neq 0\}$,*

$$\begin{aligned}
&k_F^* = \arg\min_{k_F} \|k_F - k_{F0}\|^2 \\
&s.t. \\
&k_F \geq 0 \\
&\left(1 - sgn(L_g b_F(x)_i) sgn(L_g L_f^{m-1} b(x)_i)\right)\left(k_F \alpha_F(b_F(x)) + L_f b_F(x) - \frac{L_g b_F(x)_i (L_f^m b(x) + S(b(x)) + \alpha_m(\psi_{m-1}(x)))}{L_g L_f^{m-1} b(x)_i}\right) \geq 0, \\
&k_F \alpha_F(b_F(x)) + L_f b_F(x) - \sum_{k=1}^{q} \min(-L_g b_F(x)_k u_{min,k}, -L_g b_F(x)_k u_{max,k}) \geq 0
\end{aligned} \tag{22}$$

then the QP (20) is feasible. k_{F0} is the initial value of k_F. $L_g b_F(x)_k$ is the kth component of $L_g b_F(x)$.

Proof. To prove the feasibility of QP (20), it is necessary to prove that all constraints in QP (20) are compliant with each other, that is, the $K_s(x) \cap K_F(x) \cap U \neq \emptyset$. Noted that the set $K_F(x)$ and $K_s(x)$ formulate two half spaces and the boundaries of sets are hyperplanes. Firstly, for the feasibility constraint, the hyperplane of half space formed by $K_F(x)$ can be decided via intercepts l_j *, $j \in \{k \in \mathbb{R} : L_g b_F(x)_k \neq 0\}$,

$$l_j = \frac{L_f b_F(x) + k_F \alpha_F(b_F(x))}{-L_g b_F(x)_j}. \tag{23}$$

The feasibility constraint makes control input satisfy the following condition:

$$\begin{cases} u_j \geq l_j, & \text{if } -L_g b_F(x)_j < 0, \\ u_j \leq l_j, & \text{if } -L_g b_F(x)_j > 0. \end{cases} \tag{24}$$

According to the derivation of feasibility constraint, the last constraint in QP (22) make $K_F(x) \cap U \neq \emptyset$.

Secondly, for the safety constraint, the hyperplane of half space formed by $K_s(x)$ can be decided via intercepts $d_j, j \in \{k \in \mathbb{R} : L_g L_f^{m-1} b(x)_k \neq 0\}$ (e.g., in Figure 1).

$$d_j = \frac{L_f^m b(x) + S(b(x)) + \alpha_m(\psi_{m-1}(x))}{-L_g L_f^{m-1} b(x)_j}. \tag{25}$$

The safety constraint makes control input satisfy the following condition:

$$\begin{cases} u_j \geq d_j, & \text{if } -L_g L_f^{m-1} b(x)_j < 0, \\ u_j \leq d_j, & \text{if } -L_g L_f^{m-1} b(x)_j > 0. \end{cases} \tag{26}$$

The feasibility constraint makes $K_s(x) \cap U \neq \emptyset$.

Based on the above conditions, when all components of $L_g b_F(x)$ and $L_g L_f^{m-1} b(x)$ satisfy $sgn(L_g b_F(x)_i) sgn(L_g L_f^{m-1} b(x)_i) = 1$ (e.g., in Figure 1a), the last constraint in QP (22) make at least the control input $u = [0, \dots, \min(d_i, l_i, u_{max,i}), \dots, 0]$ belong to $K_s(x) \cap K_F(x) \cap U$ for $L_g b_F(x)_i < 0$ and $u = [0, \dots, \max(d_i, l_i, u_{min,i}), \dots, 0]$ belong to $K_s(x) \cap K_F(x) \cap U$ for $L_g b_F(x)_i > 0$. So, $K_s(x) \cap K_F(x) \cap U \neq \emptyset$ is guaranteed in this case.

When $sgn(L_g b_F(x)_i) sgn(L_g L_f^{m-1} b(x)_i) = -1$ (e.g., in Figure 1b), the feasibility constraint and safety constraint is conflict-free, if all intercepts d_i satisfy

$$\begin{cases} d_i \leq l_i, & \text{if } -L_g L_f^{m-1} b(x)_i < 0, -L_g b_F(x)_i > 0, \\ d_i \geq l_i, & \text{if } -L_g L_f^{m-1} b(x)_i > 0, -L_g b_F(x)_i < 0. \end{cases} \tag{27}$$

The condition (27) is equal to the second constraint in QP (22). In this case, at least the control input $u = [0, \dots, \hat{u}_i, \dots, 0]$ fall into $K_s(x) \cap K_F(x) \cap U$, where $\hat{u}_i$ is

$$\begin{cases} \max(d_i, u_{min,i}) \leq \hat{u}_i \leq \min(l_i, u_{max,i}), & \text{if } -L_g L_f^{m-1} b(x)_i < 0, -L_g b_F(x)_i > 0, \\ \max(l_i, u_{min,i}) \leq \hat{u}_i \leq \min(d_i, u_{max,i}), & \text{if } -L_g L_f^{m-1} b(x)_i > 0, -L_g b_F(x)_i < 0. \end{cases} \tag{28}$$

That is $K_s(x) \cap K_F(x) \cap U \neq \emptyset$ is guaranteed in this case.

In summary, the QP (22) can make all constraints conflict-free at the same time, that is, $K_s(x) \cap K_F(x) \cap U \neq \emptyset$, because at least the control input $[0, \dots, u_z, \dots, 0] \in K_s(x) \cap K_F(x) \cap U, z \in [1, \dots, q]$ can be found by QP (22). The control component u_z is

$$\begin{cases} u_z = \hat{u}_z, & \text{if } sgn(L_g b_F(x)_z) sgn(L_g L_f^{m-1} b(x)_z) = -1, \\ u_z = \min(d_z, l_z, u_{max,z}), & \text{if } L_g L_f^{m-1} b(x)_z < 0, L_g b_F(x)_z < 0, \\ u_z = \max(d_z, l_z, u_{min,z}), & \text{if } L_g L_f^{m-1} b(x)_z > 0, L_g b_F(x)_z > 0, \\ u_z = \min(d_z, u_{max,z}), & \text{if } L_g b_F(x)_z = 0, L_g L_f^{m-1} b(x)_z < 0, \\ u_z = \max(d_z, u_{min,z}), & \text{if } L_g b_F(x)_z = 0, L_g L_f^{m-1} b(x)_z > 0, \\ u_z = \min(l_z, u_{max,z}), & \text{if } L_g L_f^{m-1} b(x)_z = 0, L_g b_F(x)_z < 0, \\ u_z = \max(l_z, u_{min,z}), & \text{if } L_g L_f^{m-1} b(x)_z = 0, L_g b_F(x)_z > 0. \end{cases} \tag{29}$$

■

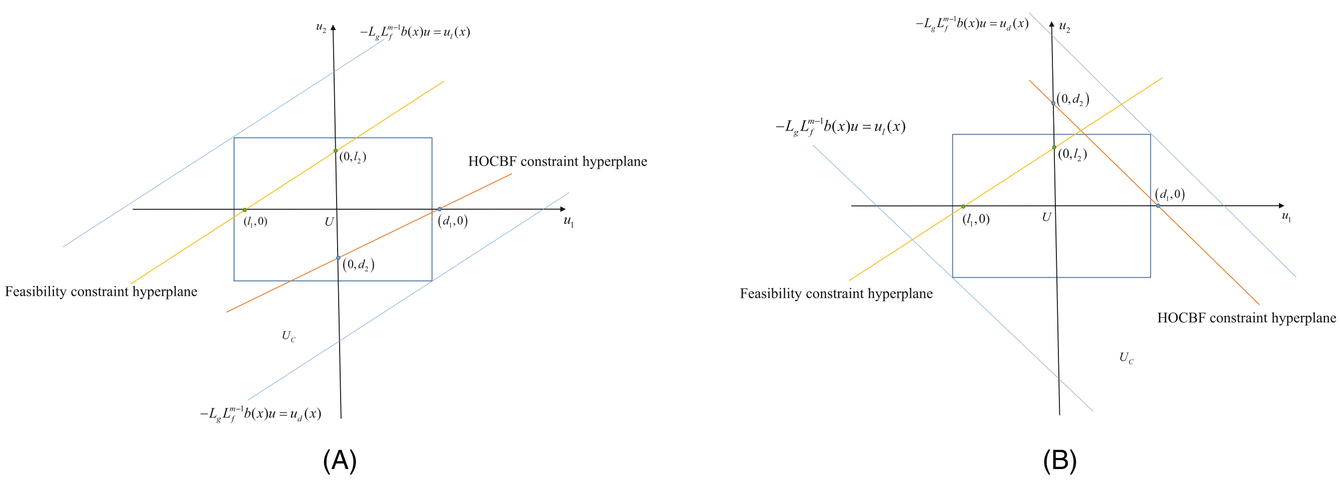

FIGURE 1 The relationship between HOCBF constraint hyperplanes and feasibility constraint hyperplanes in the case of a two dimensional control $u = [u_1, u_2]$. (a) All components of $L_g b_F(x)$ and $L_g L_f^{m-1} b(x)$ satisfy $sgn(L_g b_F(x)_i) sgn(L_g L_f^{m-1} b(x)_i) = 1$. (b) There exits a component of $L_g b_F(x)$ and $L_g L_f^{m-1} b(x)$ satisfy $sgn(L_g b_F(x)_i) sgn(L_g L_f^{m-1} b(x)_i) = -1$.

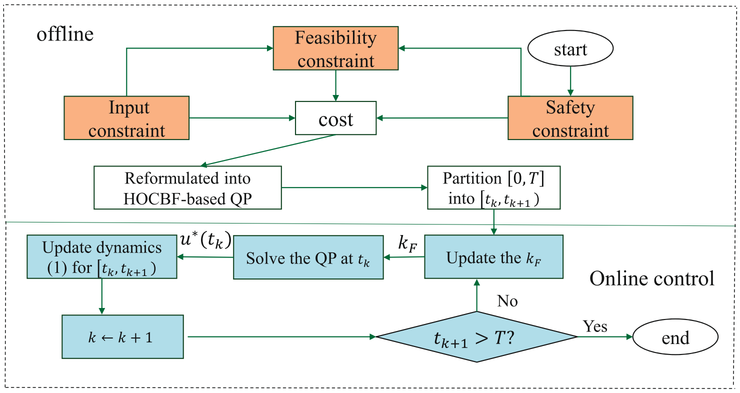

FIGURE 2 The process of solving the safety-critical control problem with feasibility-guaranteed method.

Remark 2. The QP (20) enforces $b_F(x) \geq 0$. When $b_F(x) > 0$, all constraints in QP (22) are the form of $k_F \geq \frac{a}{\alpha_F(b_F(x))}, a \in \mathbb{R}$ so that all constraints compliant with each other. Therefore, the QP (22) is always feasible.

However, when $b_F(x) = 0$, the QP (22) is invalid such that the feasibility of QP (20) can not be guaranteed. Then, it can make $b_F(x) > 0$ by changing the class $\mathcal{K}$ function in HOCBF as follows:

$$L_f^m b(x) + L_g L_f^{m-1} b(x) u + S(b(x)) + \delta \alpha_m(\psi_{m-1}(x)) \geq 0, \tag{30}$$

where $\delta > 1$ if $b_F(x) = 0$. If $\psi_{m-1}(x) \neq 0$, $b_F(x) > 0$ since $\delta \alpha_m(\psi_{m-1}(x)) > \alpha_m(\psi_{m-1}(x))$. Meanwhile, consider that the function $\psi_{m-1}(x)$ may be 0, it can make $b_F(x) > 0$ by changing the class $\mathcal{K}$ functions in function $\psi_i(x), i = [1, \ldots, m-1]$ as follows:

$$\psi_i(x) = \dot{\psi}_{i-1}(x) + \delta_i \alpha_i(\psi_{i-1}(x)), \tag{31}$$

when the vector $[\psi_{m-1}(x), \dot{\alpha}_{m-1}(\psi_{m-2}), \ldots, \dot{\alpha}_{i+1}(\psi_i)] = 0$. If $\dot{\alpha}_i(\psi_{i-1}) > 0$, $\delta_i > 1$, otherwise, $0 < \delta_i < 1$. Then the condition $b_F(x) > 0$ is satisfied such that the QP (22) can be used to guarantee the feasibility of QP (20).

Now, the feasibility problem of QP (20) is solved by Theorem 1. The process of solving the safety-critical control problem with feasibility-guaranteed method is shown in Figure 2.

Remark 3. Compared to[14] and[17], this method has two advantages. Firstly, it remains effective even when $b(x) = 0$, whereas the approach presented in Reference 14 may not guarantee feasibility through alterations in the decay rate within the HOCBF constraint. Secondly, the feasibility constraint in our method is readily obtained and can be directly incorporated into QP (20), eliminating the need to seek a candidate function as required in Reference 17

3.2 | Feasibility of QP with multiple HOCBFs

When the system has multiple safety constraints, it is challenging to guarantee the feasibility of QP with multiple HOCBF constraints in advance. It requires that all constraints cannot conflict with each other, nor can these constraints conflict with input constraint. To guarantee the feasibility of QP, the Type-2 HOCBF is proposed, which has less restrictive than HOCBF. It effectively simplifies the QP with multiple HOCBF constraints by consolidating them into a single HOCBF constraint.

Generally, the HOCBF condition should hold for $x \in C_1$, while the Type-2 HOCBF condition only needs hold for $x \in A \subset C_1$. To give the definition of Type-2 HOCBF, the associated set A for Type-2 HOCBF is defined.

$$A = \{x \in \mathbb{R}^n : 0 \leq b(x) \leq a\}, \tag{32}$$

Then, the definition of Type-2 HOCBF is given as below.

Definition 8. A function $b(x) : \mathbb{R}^n \to \mathbb{R}$ is a Type-2 High Order Control Barrier Function (HOCBF) of relative degree m for system (1), if there exist $(m-i)$-order differentiable class $\mathcal{K}$ functions $\alpha_i, i \in \{1, \dots, m-1\}$, a class $\mathcal{K}$ functions α_m and a set A such that

$$\sup_{u \in \mathbb{R}^q} [L_f^m b(x) + L_g L_f^{m-1} b(x) u + S(b(x)) + \alpha_m(\psi_{m-1}(x))] \geq 0, \quad \forall x \in A. \tag{33}$$

If $b(x)$ is a Type-2 HOCBF, then the set of control input can be defined as

$$K_{\bar{s}}(x) = \{u \in \mathbb{R}^q : \text{if } x \in A, \text{then (33) holds}\}. \tag{34}$$

Then, it holds the following conclusion.

Theorem 2. *For system (1) and the continuously differentiable function $b(x) : \mathbb{R}^n \to \mathbb{R}$, assume that $b(x)$ is a Type-2 HOCBF for a given set A defined by (32). If there exists a locally Lipschitz continuous control input $u \in K_{\bar{s}}$ and $x(0) \in C_1 \cap \cdots \cap C_m$, then the set $C_1 \cap \cdots \cap C_m$ is forward invariant for system (1).*

Proof. Firstly, there exists a control u satisfying (33), so $K_{\bar{s}}$ is non-empty. For $x(0) \in C_1 \cap \cdots \cap C_m$, there are two cases.

Case 1: If $x(0) \in A \cap C_2 \cap \cdots \cap C_m$, then $b(x)$ is a Type-2 HOCBF. So, $\psi_m(x) \geq 0$ holds on the boundary of C_m, that is, $\dot{\psi}_{m-1}(x) + \alpha_m(\psi_{m-1}(x)) \geq 0$. Since $x(0) \in C_m$, the Brezis' Theorem (see,[26] Theorem 1) ensures that $\psi_{m-1}(x) \geq 0$ and the states are always in the set C_m. The conditions $\psi_{m-1}(x) \geq 0$ and $x(0) \in C_{m-1}$ enforce the states $x \in C_{m-1}$ for all time and $\psi_{m-2}(x) \geq 0$. Iteratively, $x \in C_i, \forall i \in [1, \dots, m]$, so the set $C_1 \cap \cdots \cap C_m$ is forward invariant.

Case 2: If $x(0) \in \bar{a} \cap C_2 \cap \cdots \cap C_m$, then $\bar{a} = \{x \in \mathbb{R}^n : b(x) > a\}$, $b(x)$ is not a Type-2 HOCBF. If state x always satisfies $b(x) > a$ with the help of control input, then the set $C_1 \cap \cdots \cap C_m$ is forward invariant. However, if the state satisfies $b(x) \in A$ at current time t by control input, then $b(x)$ is a Type-2 HOCBF. It transforms to Case 1, which thus gets the forward invariance of $C_1 \cap \cdots \cap C_m$. ∎

When the system has N safety constraints, for example, $b_i(x) : \mathbb{R}^n \to \mathbb{R}$, $i \in [1, \dots, N]$, the associated set for each Type-2 HOCBF can be defined as follows:

$$C_j^i = \{x \in \mathbb{R}^n : \psi_{j-1}^i(x) \geq 0\}, \quad j = \{1, \dots, m\}, \tag{35}$$

$$A_i = \{x \in \mathbb{R}^n : 0 \leq b_i(x) \leq a_i\}, \tag{36}$$

$$\sup_{u \in \mathbb{R}^q} [L_f^m b_i(x) + L_g L_f^{m-1} b_i(x) u + S(b_i(x)) + \alpha_m^i(\psi_{m-1}^i(x))] \geq 0, \quad \forall x \in A_i, \tag{37}$$

and the sets of control input satisfying (37) are

$$K_{\bar{s}}^i(x) = \{u \in \mathbb{R}^q : \text{if } x \in A_i, \text{then (37) holds}\}. \tag{38}$$

For multiple Type-2 HOCBFs, assume that A_i does not overlap for every $i \in [1, \dots, N]$. Then, a control law can be given to address the multiple security constraints of the system independently. If the state $x \notin A_i$, the control input is $u = u_{nom}$. When the state x enters any set A_i, the follow QP is implemented for the associated $b_i(x)$.

$$u^* = \arg\min_u \|u - u_{nom}\|^2 \tag{39}$$

$$s.t.$$

$$L_f^m b_i(x) + L_g L_f^{m-1} b_i(x) u + S(b_i(x)) + \alpha_m^i(\psi^i(m-1)(x)) \geq 0, \tag{40}$$

$$L_f b_F^i(x) + L_g b_F^i(x) u + k_F^i b_F^i(x) \geq 0, \tag{41}$$

$$u_{min} \leq u \leq u_{max}. \tag{42}$$

This control input makes the set $C_1^i \cap \cdots \cap C_m^i$ forward invariant.

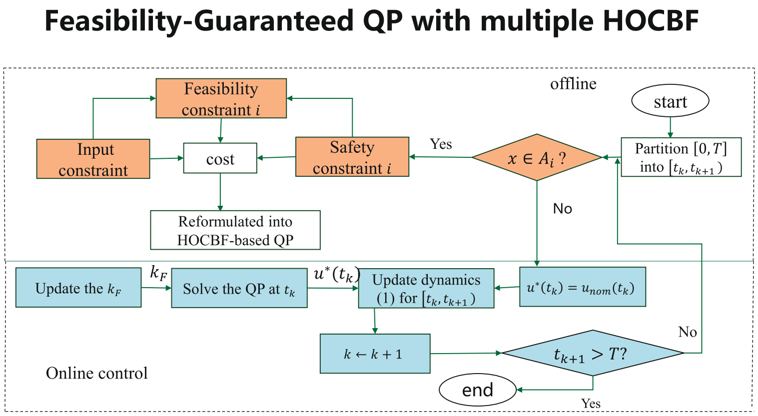

FIGURE 3 The process of safety-critical control with multiple HOCBF constraints.

In (39)-(42), a feasible constraint $b_F^i(x)$ is added in QP. According to Theorem 1, the parameter k_F^i should be updated by QP (22) to guarantee the feasibility. The process of safety-critical control with multiple HOCBF constraints is shown in Figure 3.

Remark 4. a_i, $i \in [1, \dots, N]$, can be different from each other. a_i in A_i should be small to reduce the interference of u_{nom}. Noting that if a_i is too small, the controller may be sensitive to noise.

Theorem 3. *Given the N continuously differentiable functions $b_i(x) : \mathbb{R}^n \to \mathbb{R}$, $i \in [1, \dots, N]$ for system (1), suppose that $u_{nom} \subset U$ is locally Lipschitz continuous, and $u = u_{nom}$ if $x \notin A_i$. If each $b_i(x)$ is a Type-2 HOCBF with associated u from (39)-(42), and for any $j, k \in [1, \dots, N]$, $j \neq k$, $A_j \cap A_k = \emptyset$, then $x(0) \in C^1 \cap \cdots \cap C^N$, then u ensures that the system is safe.*

Proof. Since for any $j, k \in [1, \dots, N], j \neq k, A_j \cap A_k = \emptyset$, there exists a unique i for which $x \in A_i$, when $x \in A_i$, according to Theorems 1 and 2, the control input obtained by QP (39) subject to (39)-(42) ensures that the system is safe. When x leaves A_i, for $b_i(x) > a_i$, the $u = u_{nom}$ still guarantees safety. ■

4 | APPLICATION TO 3-DOF PLANE ROBOT

The obstacle avoidance task of a 3-DOF plane robot is used to verify the effectiveness of the proposed method. In this section, the dynamic model of the robot is introduced, followed by the process of how to get a feasibility constraint.

The 3-DOF robot is shown in Figure 4, where the vector of generalized coordinates $q = [q_1, q_2, q_3]^T$ represents the joint angle, m_1, m_2, and m_3 are the mass of three links, L_{c1}, L_{c2}, L_{c3} are the distance from the respective joint axes to the centers of mass, L_1, L_2, and L_3 are the lengths of three links, and J_1, J_2, and J_3 are the moment of inertia of three links. The Euler–Lagrangian dynamics is given as follows:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = u, \tag{43}$$

where $q \in \mathbb{R}^3$, $M(q) \in \mathbb{R}^{3\times3}$, $C(q, \dot{q}) \in \mathbb{R}^{3\times3}$ are inertia matrix and Coriolis-centrifugal matrix, respectively. $M(q)$ is symmetric and positive-definite.[27] $u = [\tau_1, \tau_2, \tau_3] \in U$ is the control input.

Assuming that the robot needs to complete a trajectory while avoiding a circular obstacle located at (x_0, y_0), the end-effector must be at least a radius r away from the obstacle. Thus, the safety constraint is described by a continuously differentiable function $b(q) : \mathbb{R}^3 \to \mathbb{R}$ as follows:

$$b(q) = (x(q) - x_0)^2 + (y(q) - y_0)^2 - r^2 \geq 0, \tag{44}$$

where $x(q)$, $y(q)$ are the coordinates of end-effector in form of

$$\begin{aligned} x(q) &= -L_3 \sin(q_1 + q_2 - q_3) - L_2 \sin(q_1 + q_2) - L_1 \sin q_1, \\ y(q) &= L_3 \cos(q_1 + q_2 - q_3) + L_2 \cos(q_1 + q_2) + L_1 \cos q_1. \end{aligned} \tag{45}$$

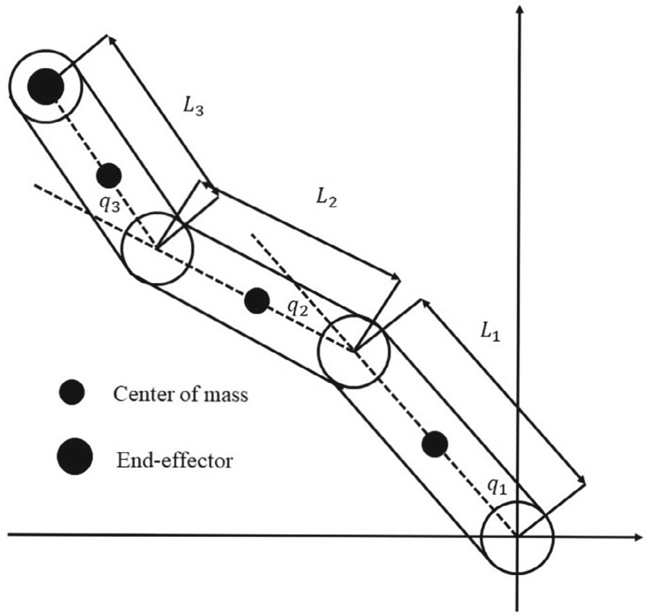

FIGURE 4 Figure shows a 3-DOF plane robotic system.

The associated safety set is defined by

$$S = \{q \in \mathbb{R}^3 : b(q) \geq 0\}. \tag{46}$$

Obviously, the relative degree of $b(q)$ is 2, so the HOCBF is used to ensure the safety of system. Define the class $\mathcal{K}$ functions $\alpha_i(\psi_{i-1}(q)) = k_i\psi_{i-1}(q)$ for $i = 1, 2$. Then, the following controller $K_s(q, \dot{q})$ guarantees the forward invariance of S

$$\begin{aligned} K_s(q, \dot{q}) = \{u \in \mathbb{R}^3 : L_f^2 b(q) + L_g L_f b(q) u \\ + (k_1 + k_2) L_f b(q) + k_1 k_2 b(q) \geq 0\}. \end{aligned} \tag{47}$$

The feasibility constraint $b_F^*(q) : \mathbb{R}^3 \to \mathbb{R}$ can be derived from HOCBF constraint and input constraint, that is,

$$\begin{aligned} b_F^*(q) = L_f^2 b(q) + (k_1 + k_2) L_f b(q) + k_1 k_2 b(q) \\ - \sum_{i=1}^{3} \min(-L_g L_f b(q)_i u_{min,i}, -L_g L_f b(q)_i u_{max,i}) \geq 0. \end{aligned} \tag{48}$$

Then, QP (20) is constructed to solve the optimal control, in which the Lie derivative of $b_F^*(q)$ along (43) needs to calculate the derivative of $M(q)^{-1}$. $M(q)$ is decided by the structural parameters of the robot, but the form of $M(q)^{-1}$ is hard to get. So, the energy-based safety constraint $b_D(q, \dot{q})$ is used instead of safety constraint $b(q)$ to solve the problem.

Definition 9 (28). Given a safety constraint $b(q) : \mathbb{R}^3 \to \mathbb{R}$, and the corresponding safe set S, the associated energy-based safety constraint and corresponding energy-based safe set are defined as follows:

$$b_D(q, \dot{q}) = -\frac{1}{2}\dot{q}^T M(q) \dot{q} + \alpha_e b(q), \tag{49}$$

$$S_D = \{(q, \dot{q}) \in \mathbb{R}^3 \times \mathbb{R}^3 : b_D(q, \dot{q}) \geq 0\} \tag{50}$$

where $\alpha_e > 0$.

According to Proposition 1 in Reference 28, S and S_D satisfy

$$S_D \subset S, \mathrm{Int}(S) \subset \lim_{\alpha_e \to \infty} S_D \subset S, \tag{51}$$

where S is the limit of S_D when α_e is sufficiently large. So, the safety-critical controller only needs to ensure S_D is forward invariant.

Remark 5. The velocity term in $b_D(q, \dot{q})$ acts to moderate the speed at which the system approaches the boundary of S. If the systems with low inertia can approach the boundary at higher speeds, they can be slowed down more easily.

According to the fact that $\dot{M}(q) - 2C(q, \dot{q})$ is skew-symmetric, $\dot{q}^T(\dot{M}(q) - 2C(q, \dot{q}))\dot{q} = 0$ can be get. The Lie derivative of $b_D(q, \dot{q})$ along (43) is

$$\begin{aligned} L_f b_D(q, \dot{q}) + L_g b_D(q, \dot{q})u &= -\dot{q}^T M(q)\ddot{q} - \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} + \alpha_e \dot{b}(q) \\ &= -\dot{q}^T u - \frac{1}{2}[\dot{q}^T(\dot{M}(q) - 2C(q, \dot{q}))\dot{q}] + \alpha_e \dot{b}(q) \\ &= \alpha_e L_f b(q) - \dot{q}^T u \end{aligned} \tag{52}$$

When $b_D(q, \dot{q})$ is the safety constraint, the controller $K_D(q, \dot{q})$ makes the set S_D forward invariant.

$$K_D(q, \dot{q}) = \{u \in \mathbb{R}^3 : L_f b_D(q, \dot{q}) + L_g b_D(q, \dot{q})u + \alpha(b_D(q, \dot{q})) \geq 0\}. \tag{53}$$

Then, the corresponding feasibility constraint $b_F(q, \dot{q})$ is

$$b_F(q, \dot{q}) = \alpha_e L_f b(q) + \alpha(b_D(q, \dot{q})) + \sum_{i=1}^{3} \min(\dot{q}_i u_{min,i}, \dot{q}_i u_{max,i}), \tag{54}$$

whose relative degree is one and Lie derivative is easy to get. Finally, the following QP is used to get the control input at current time.

$$\begin{aligned} & u^* = \arg\min_u \|u - u_{nom}\|^2 \\ & s.t. \\ & L_f^2 b(q) + L_g L_f b(q)u + (k_1 + k_2)L_f b(q) + k_1 k_2 b(q) \geq 0, \\ & L_f b_F(q, \dot{q}) + L_g b_F(q, \dot{q})u + k_F \alpha_F(b_F(q, \dot{q})) \geq 0, \\ & u_{min} \leq u \leq u_{max}, \end{aligned} \tag{55}$$

In order to guarantee the feasibility of (55), the k_F in $\alpha_F(b_F(q, \dot{q}))$ is updated by (22).

Remark 6. It is obvious that the safety constraint using $b(q) \geq 0$ is less conservative than using $b_D(q, \dot{q})$. Since $S_D \subset S$ leads to $K_D(q, \dot{q}) \subset K_s(q, \dot{q})$ and the feasibility constraint $b_F(q, \dot{q})$ guarantees $K_D(q, \dot{q}) \cap U \neq \emptyset$, the feasibility constraint $b_F(q, \dot{q})$ is derived from $b_D(q, \dot{q})$. It still guarantees that $b(q) \geq 0$ does not conflict with the control constraint.

If there are multiple circular obstacles centered at (x_i, y_i), $i \in \{1, \ldots, N\}$ on the trajectory and the obstacles do not overlap, then the system has N safety constraints. Then, Type-2 HOCBF is used to enforce the safety. For the ith safety constraint, the Type-2 HOCBF and associated safety sets are defined as follows:

$$b_i(q) = (x(q) - x_i)^2 + (y(q) - y_i)^2 - r_i^2, \tag{56}$$

$$A_i = \left\{ q \in \mathbb{R}^3 : b_i(q) \in \left[0, \frac{1}{2}\dot{q}^T M(q)\dot{q}\right] \right\}. \tag{57}$$

where r_i is the radius of ith obstacle. Then, the feasibility constraints $b_F^i(q, \dot{q})$ are decided by the energy-based safety constraint $b_D^i(q, \dot{q})$.

$$b_D^i(q,\dot{q}) = -\frac{1}{2}\dot{q}^T M(q)\dot{q} + \alpha_e^i b_i(q,\dot{q}), \tag{58}$$

$$b_F^i(q,\dot{q}) = \alpha_e^i L_f b_i(q) + \alpha_i(b_D^i(q,\dot{q})) + \sum_{i=1}^{3} \min(\dot{q}_i u_{min,i}, \dot{q}_i u_{max,i}). \tag{59}$$

If the state $q \notin A_i$, $u = u_{nom}$. Otherwise, the control input is obtained by following QP and its feasibility is guaranteed by updating k_F^i.

$$\begin{aligned} &u^* = \arg\min_u \|u - u_{nom}\|^2 \\ &s.t. \\ &L_f^2 b_i(q) + L_g L_f b_i(q) u + (k_1^i + k_2^i) L_f b_i(q) + k_1^i k_2^i b(q) \geq 0, \\ &L_f b_F^i(q,\dot{q}) + L_g b_F^i(q,\dot{q}) u + k_F^i \alpha_F^i (b_F(q,\dot{q})) \geq 0, \\ &u_{min} \leq u \leq u_{max}, \end{aligned} \tag{60}$$

The proposed control law has two advantages. Firstly, it guarantees that multiple safety constraints and input constraint are met simultaneously. Secondly, through the transformation of the QP to include a single safety constraint using Type-2 HOCBF, the computational efficiency of the QP remains unaffected even as the number of safety constraints increases.

5 | NUMERICAL RESULTS

In this section, the simulation results are given. Two cases are included in the simulation studies: Case 1—there is one obstacle on the desired trajectory, Case 2—there are two obstacles on the desired trajectory.

The structural parameters of the robot are shown in Table 1. The nominal controller is a PID controller. The control input u is bounded by $u_{min} = [-5, -5, -5]Nm$ and $u_{max} = [5, 5, 5]Nm$, $q(0) = [0.05, 0, 0]rad$. The quadprog is used to solve the QPs.

In the first case, the obstacle is centered at $(-1, 2)$ with the radius $r = 0.1$, $\alpha_e = 1$, $k_{F0} = 1.85$, and the class $\mathcal{K}$ function in safety constraint is $k_1 = 14$, $k_2 = 4$. In the second case, the obstacles are respectively centered at $(-1, 2)$ and $(-0.6, 2.2)$ with the radius $r_1 = r_2 = 0.1$. The parameters of energy-based safety constraints are $\alpha_e^1 = 1$, $\alpha_e^2 = 2$. $k_{F0}^1 = 1.85$, $k_{F0}^2 = 1$, and the class $\mathcal{K}$ functions in the two safety constraints are chosen as $k_1^1 = 14$, $k_2^1 = 4$, $k_1^2 = 15$, $k_2^2 = 1.5$.

The trajectory tracking performance is shown in Figure 5. Figure 5a,b give the trajectories of robots end-effector with and without safety constraint, respectively. As shown in Figure 5, the robot with safety constraints control avoids obstacle area. Figure 6 shows the response curves of function $b_i(x)$ and $\psi_1^i(x)$, which are all greater than 0, indicating the forward invariance of the set $C_1^i \cap C_2^i$, $i \in \{1, 2\}$. The control input with single safety constraint is shown in Figure 7. As shown in Figure 7a, QPs are feasible with the proposed method. As a comparison, from Figure 7b, if there is no feasibility constraint, QPs become infeasible. It is in accordance with the analysis of Theorem 1. Figure 8 shows the control inputs with two obstacles on the desired trajectory.

To demonstrate the adaptability of the proposed method with multiple safety constraints to various input constraints, another four comparison studies are also given, which correspond to case-$u_{min} = -5$, case-$u_{min} = -4$, case-$u_{min} = -3$, and the case-time-varying input constraint. The results are presented in Figure 9, revealing that in all four cases, the robot effectively avoids obstacles, and QPs remain feasible.

TABLE 1 Structural parameters of robot.

Joint i	$m_i/(kg)$	$L_i/(m)$	$L_{ci}/(m)$	$J_i/(kg \cdot m^2)$
1	1.0	1.0	0.5	0.0833
2	0.8	0.8	0.4	0.0427
3	0.6	0.6	0.3	0.0180

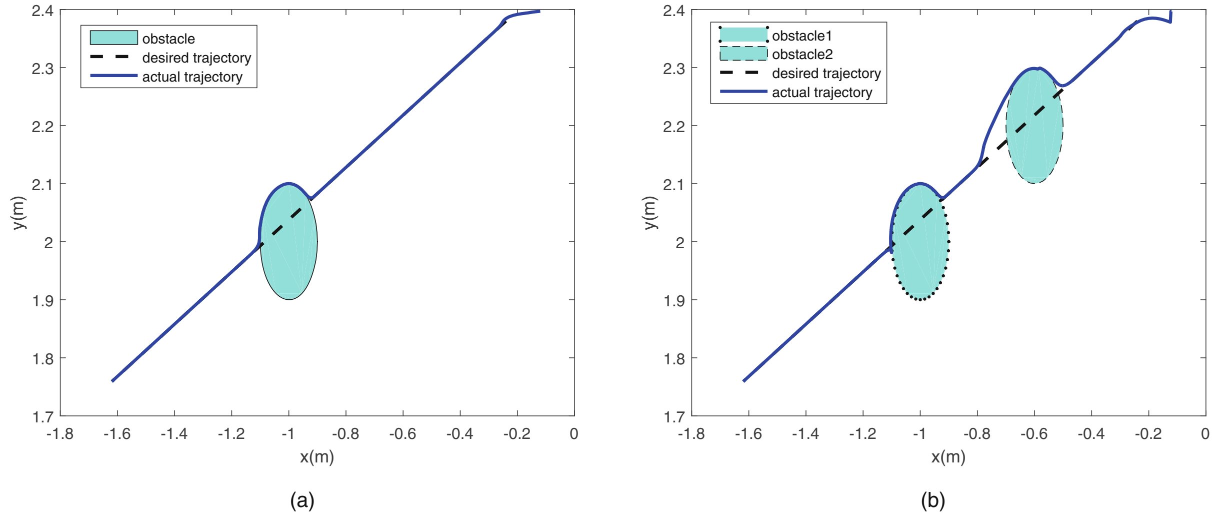

FIGURE 5 Trajectories of robot end-effector. (a) One-obstacle case. (b) Two-obstacle case.

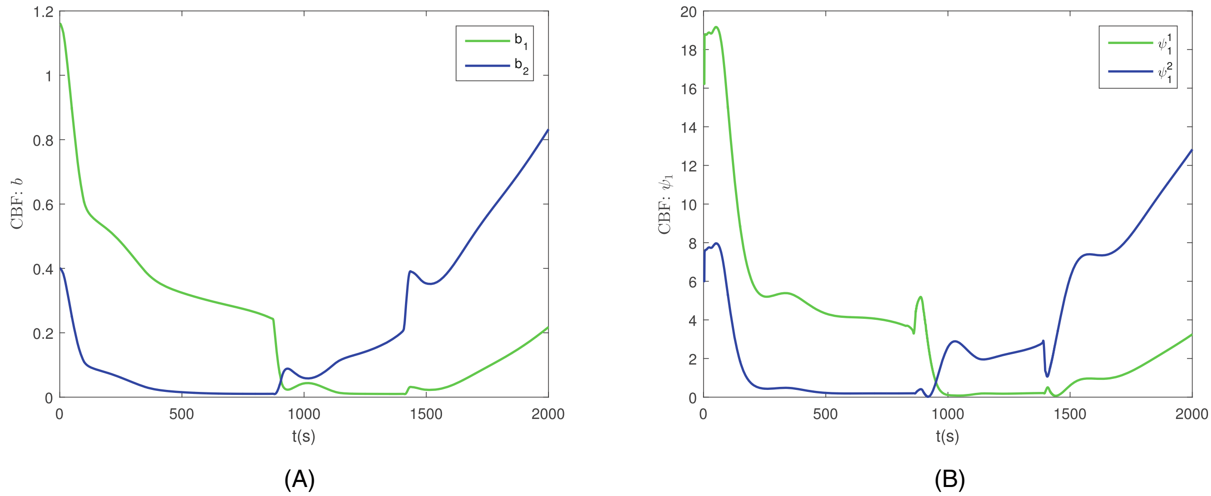

FIGURE 6 Response curves of $b(x)$ and $\psi_1(x)$. (a) $b_1(x)$ and $b_2(x)$. (b) $\psi_1^1(x)$ and $\psi_1^2(x)$.

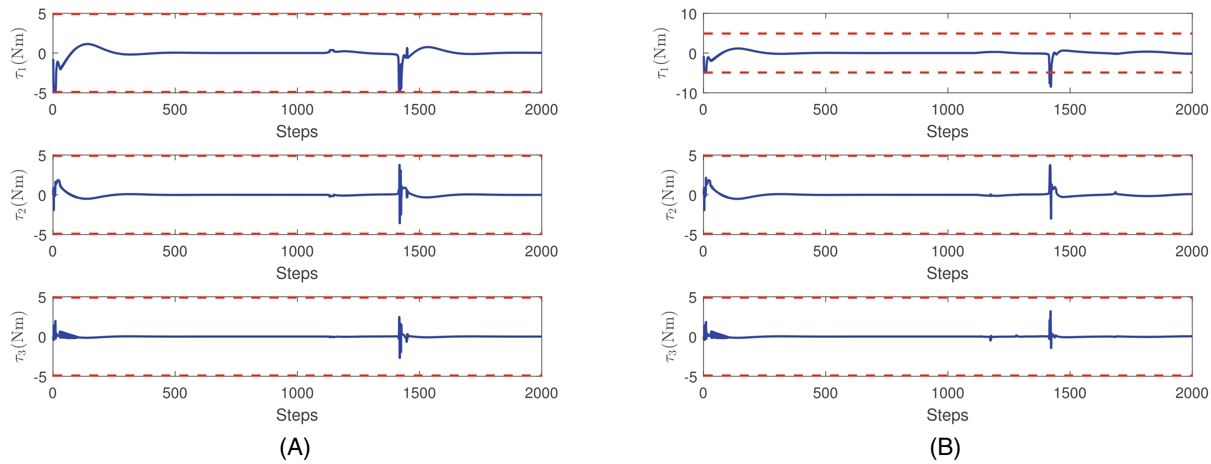

FIGURE 7 Control inputs with one obstacle on the desired trajectory. (a) Control inputs of the proposed method. (b) Control inputs without feasibility constraint.

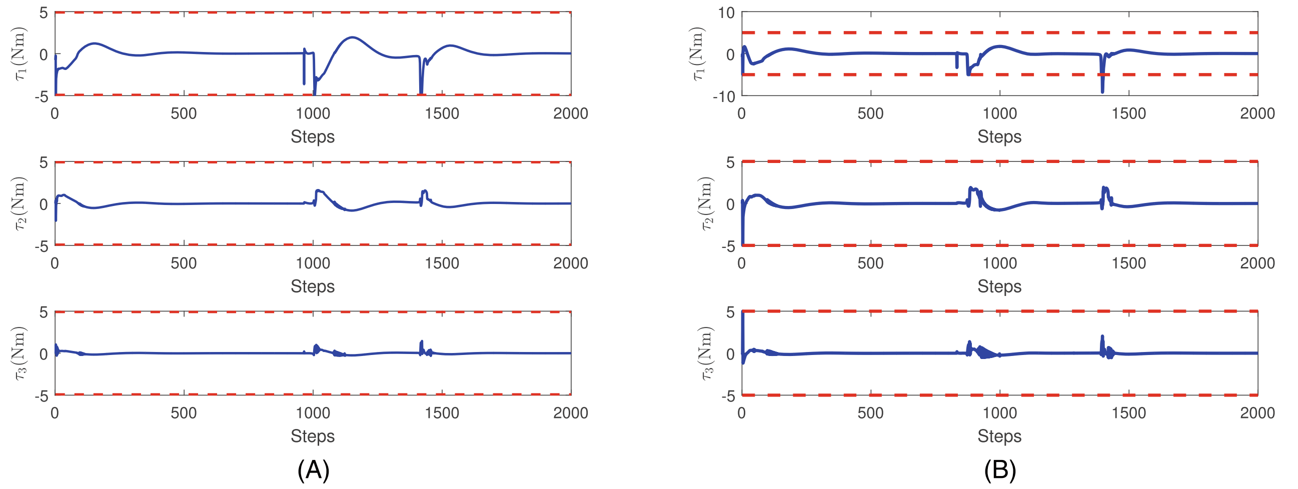

FIGURE 8 Control inputs with two obstacles on the desired trajectory. (a) Control inputs of the proposed method. (b) Control inputs without feasibility constraint.

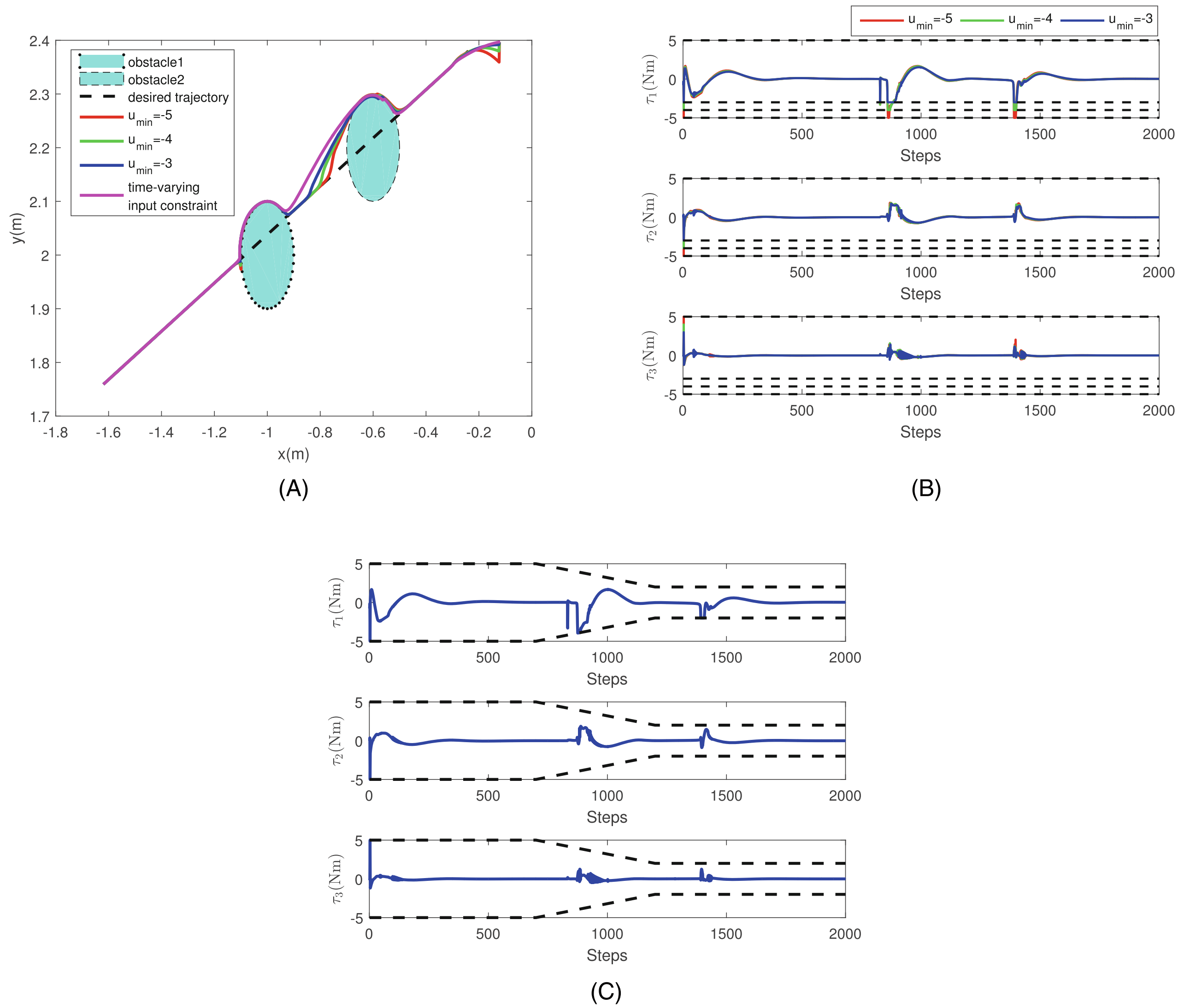

FIGURE 9 Control performance of the proposed method under different input constraints. (a) Trajectories of the robot end effector. (b) Control signals with fixed input constraints. (c) Control signals with time-varying input constraint.

6 | CONCLUSION

In this article, a feasibility-guaranteed QPs is proposed. Firstly, the QP is added with a feasibility constraint defined by the input constraint and the HOCBF constraint. Then, the parameter of feasibility constraint is updated by a new QP, which is obtained by using the control sharing property. The proposed method enforces all constraints not in conflict such that the feasibility of QP guaranteed. Moreover, when the system has multiple HOCBF constraints, the Type-2 HOCBF is proposed, which makes the system constrained by a single HOCBF at current time step. Finally, the safety-critical controller is derived by using the proposed feasibility-guaranteed QP. The effectiveness is demonstrated on obstacle avoidance of a 3-DOF plane robot. The future work is to find a less conservative method for specific systems and guarantee the feasibility of QP subject to multiple intersecting constraints.

DATA AVAILABILITY STATEMENT

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

FUNDING INFORMATION

This work was supported by the National Natural Science Foundation of China under Grants 62173035, 61803033 and 61836001. And Xiaomi Young Scholars from Xiaomi Foundation.

ENDNOTE

*An example of a two dimensional control $u = [u_1, u_2]$ is shown in Figure 1.

ORCID

Yuanqing Xia https://orcid.org/0000-0002-5977-4911

REFERENCES

1. Ames AD, Coogan S, Egerstedt M, Notomista G, Sreenath K, Tabuada P. Control barrier functions: theory and applications. *European Control Conference*. IEEE; 2019:3420-3431.
2. Mayne DQ, Rawlings JB, Rao CV, Scokaert PO. Constrained model predictive control: stability and optimality. *Automatica*. 2000;36(6):789-814.
3. Tee KP, Ge SS, Tay EH. Barrier Lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*. 2009;45(4):918-927.
4. Ames AD, Grizzle JW, Tabuada P. Control barrier function based quadratic programs with application to adaptive cruise control. *IEEE Conference on Decision and Control*. IEEE; 2014:6271-6278.
5. Wieland P, Allgöwer F. Constructive safety using control barrier functions. *IFAC Proc Vol*. 2007;40(12):462-467.
6. Hsu SC, Xu X, Ames AD. Control barrier function based quadratic programs with application to bipedal robotic walking. *American Control Conference*. IEEE; 2015:4542-4548.
7. Nguyen Q, Sreenath K. Exponential control barrier functions for enforcing high relative-degree safety-critical constraints. *American Control Conference*. IEEE; 2016:322-328.
8. Xiao W, Belta C. Control barrier functions for systems with high relative degree. *IEEE Conference on Decision and Control*. IEEE; 2019:474-479.
9. Ames AD, Galloway K, Sreenath K, Grizzle JW. Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics. *IEEE Trans Autom Contr*. 2014;59(4):876-891.
10. Ames AD, Xu X, Grizzle JW, Tabuada P. Control barrier function based quadratic programs for safety critical systems. *IEEE Trans Autom Contr*. 2016;62(8):3861-3876.
11. Galloway K, Sreenath K, Ames AD, Grizzle JW. Torque saturation in bipedal robotic walking through control Lyapunov function-based quadratic programs. *IEEE Access*. 2015;3:323-332.
12. Xiao W, Belta C. High order control barrier functions. *IEEE Trans Autom Contr*. 2021;67(7):3655-3662. doi:10.1109/TAC.2021.3105491
13. Xiao W, Belta C, Cassandras CG. Adaptive control barrier functions. *IEEE Trans Autom Contr*. 2021;67(5):2267-2281. doi:10.1109/TAC.2021.3074895
14. Zeng J, Zhang B, Li Z, Sreenath K. Safety-critical control using optimal-decay control barrier function with guaranteed point-wise feasibility. *American Control Conference*. IEEE; 2021:3856-3863.
15. Liu S, Zeng J, Sreenath K, Belta CA. Iterative convex optimization for model predictive control with discrete-time high-order control barrier functions. *American Control Conference*. IEEE; 2023:3368-3375.
16. Chen Y, Jankovic M, Santillo M, Ames AD. Backup control barrier functions: formulation and comparative study. *IEEE Conference on Decision and Control*. IEEE; 2021:6835-6841.
17. Xiao W, Belta CA, Cassandras CG. Sufficient conditions for feasibility of optimal control problems using control barrier functions. *Automatica*. 2022;135:109960.

18. Cortez WS, Dimarogonas DV. *Safe -by-Design Control for Euler-Lagrange Systems*. Automatica. 2022;146:110620.
19. Cortez WS, Tan X, Dimarogonas DV. A robust, multiple control barrier function framework for input constrained systems. *IEEE Contr Syst Lett*. 2021;6:1742-1747.
20. Breeden J, Panagou D. *Compositions of Multiple Control Barrier Functions Under Input Constraints*. American Control Conference. IEEE; 2023:3688-3695.
21. Blanchini F. Set invariance in control. *Automatica*. 1999;35(11):1747-1767.
22. Khalil HK. *Nonlinear Systems*. Prentice-Hall; 2002.
23. Cortez WS, Oetomo D, Manzie C, Choong P. Control barrier functions for mechanical systems: theory and application to robotic grasping. *IEEE Trans Contr Syst Technol*. 2019;29(2):530-545.
24. Breeden J, Garg K, Panagou D. Control barrier functions in sampled-data systems. *IEEE Contr Syst Lett*. 2021;6:367-372.
25. Xu X. Constrained control of input–output linearizable systems using control sharing barrier functions. *Automatica*. 2018;87:195-201.
26. Brezis H. On a characterization of flow invariant sets. *Commun Pure Appl Math*. 1970;23(2):261-263.
27. Sciavicco L, Siciliano B. *Modelling and control of robot manipulators*. Springer Science & Business Media; 2001.
28. Singletary A, Kolathaya S, Ames AD. Safety-critical kinematic control of robotic systems. *IEEE Contr Syst Lett*. 2021;6:139-144.

How to cite this article: Zhang S, Zhai DH, Xiong Y, Xia Y. Feasibility-guaranteed safety-critical control with high-order control barrier function method. *Int J Robust Nonlinear Control*. 2024;34(5):3425-3441. doi: 10.1002/rnc.7141