Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 53-2 (2020) 9037–9044

Task-Priority Control of Redundant Robotic Systems using Control Lyapunov and Control Barrier Function based Quadratic Programs ⋆

Erlend A. Basso * **Kristin Y. Pettersen** *

* *Centre for Autonomous Marine Operations and Systems (NTNU AMOS), Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway (e-mail: erlend.a.basso@ntnu.no).*

Abstract: This paper presents a novel task-priority control framework for redundant robotic systems based on a hierarchy of control Lyapunov function (CLF) and control barrier function (CBF) based quadratic programs (QPs). The proposed method guarantees strict priority among different groups of tasks such as safety-related, operational and optimization tasks. Moreover, a soft priority measure in the form of penalty parameters can be employed to prioritize tasks at the same priority level. As opposed to kinematic control schemes, the proposed framework is a holistic approach to control of redundant robotic systems, which solves the redundancy resolution, dynamic control and control allocation problems simultaneously. Numerical simulations of a hyper-redundant articulated intervention autonomous underwater vehicle (AIAUV) is presented to validate the proposed framework.

Keywords: Motion control systems, mobile robots, robotics technology, optimization, nonlinear control, Lyapunov methods, redundancy resolution, task-priority control

1. INTRODUCTION

A robotic system is kinematically redundant when it has more degrees of freedom (DOFs) than those strictly required to execute a given task. Tasks are often divided into groups according to their priority, such as safety-related tasks, operational tasks and optimization tasks. Kinematic redundancy enables additional tasks to be executed simultaneously by utilizing the redundant DOFs of the system. However, care must be taken when resolving kinematic redundancy. In particular, if compatibility between two or more tasks cannot be guaranteed at all time, then lower-priority tasks may prevent convergence of higher-priority tasks. Since tasks at the lowest-priority level are often added to ensure joint-space stability of a robotic system, they are inherently incompatible with mission-related tasks such as end-effector control. Consequently, redundancy must be resolved by ensuring strict-priority between tasks, i.e. that lower-priority tasks have no effect on the execution of higher-priority tasks.

Kinematic task-priority control is a redundancy resolution method introduced in Hanafusa et al. (1981), developed in Nakamura et al. (1987) and generalized to any number of priority levels in Siciliano and Slotine (1991). This control approach decouples the controller into a kinematic and dynamic controller, and has been successfully implemented on a number of robotic systems. The framework was extended to support tasks described by sets or inequalities in Moe et al. (2016), Simetti and Casalino (2016) and Kanoun et al. (2011). These kinematic control approaches all resolve redundancy at the velocity level by generating velocity references for some dynamic controller to follow. An immediate drawback is that acceleration references cannot be included, resulting in poorer tracking accuracy.

Operational space control (Khatib, 1987) is a holistic approach that assigns joint torques directly by transforming the equations of motion from joint space into the operational space (also known as task space). Although it was mainly introduced for non-redundant systems, a dynamically consistent null space operator was defined in Khatib (1987), that allowed two operational space tasks to be defined and controlled simultaneously. In Sentis and Khatib (2004), the scheme was extended to a task-priority framework with an arbitrary number of tasks by generalizing the dynamically consistent null space operator from Khatib (1987) to an arbitrary number of priority levels. These null space operators ensure that torques generated by lower-priority tasks do not generate accelerations that affect the task dynamics of higher-priority tasks. The operational space framework was extended to include set-based tasks, i.e. control objectives with a range of valid values in Mansard et al. (2009), but this approach does not scale well for systems with a high number of DOFs.

Control Lyapunov functions (CLFs) extend Lyapunov theory to systems with inputs and have become an essential part of nonlinear control design after the pioneering work in Artstein (1983); Sontag (1983, 1989). The CLF concept

⋆ This research was partly funded by the Research Council of Norway through the Centres of Excellence funding scheme, project no. 223254 NTNU AMOS.

10.1016/j.ifacol.2020.12.2024

was extended to rapidly exponentially stabilizing control Lyapunov functions (RES-CLFs) in Ames et al. (2014a), which achieve exponential convergence at a controllable rate. Through CLFs or RES-CLFs, the control designer is free to chose among an infinite number of controllers. An important example is the point-wise minimum norm controller (Freeman and Kokotović, 1996; Petersen and Barmish, 1987), which selects the control value of minimum norm from all control values rendering the time derivative of the CLF negative definite. The point-wise minimum norm controller has a closed-form solution since it is the solution to a quadratic program (QP) with only one inequality constraint. This QP can be augmented with control input saturation limits and other control input constraints, at the expense of a closed-form solution (Galloway et al., 2015). For redundant robotic systems, two control tasks can be satisfied simultaneously by defining CLFs for each task and finding a control input that minimizes some quadratic objective function while ensuring that the time derivatives of the CLFs are negative definite (Ames and Powell, 2013). However, strict priority between tasks cannot be ensured.

Barrier functions have been used extensively in constrained optimization (Forsgren et al., 2002; Boyd and Vandenberghe, 2004), and have motivated the concept of barrier certificates for safety-critical control. Barrier certificates were introduced as a tool for proving forward invariance of sets (Prajna and Jadbabaie, 2004; Prajna, 2006). Since these sets often encoded safety-related objectives, proving invariance of a safe set implies that the system will remain safe, as long as you start safe. These barrier certificates tend to infinity as the state tends to the boundary of the safe set, and in order to obtain safety guarantees beyond the boundary of the safe set, various Lyapunov-like approaches have been proposed such as (Tee et al., 2009), where a positive definite barrier certificate is employed as a barrier Lyapunov function. However, these conditions are overly conservative since the positive definiteness property enforces the invariance of every level set, and not just the safe set of the set-based task in question.

Barrier certificates were extended to systems with inputs by introducing the first notion of a control barrier function (CBF) in Wieland and Allgöwer (2007). These control barrier functions were combined with control Lyapunov functions in Romdlony and Jayawardhana (2014), and further improved in Romdlony and Jayawardhana (2016) to establish conditions for so-called control Lyapunov-barrier functions, which jointly guarantee safety and stability. However, these conditions were shown to be too restrictive, and subsequently relaxed in Ames et al. (2014b, 2017), which extended control barrier functions to the entire safe set, and thus enabling controller synthesis through optimization-based methods (Ames et al., 2019). In particular, the CLF-based QPs in Ames and Powell (2013) and Galloway et al. (2015) could be augmented with CBFs to ensure stability and safety (Ames et al., 2014b, 2017). CBFs were generalized to exponential control barrier functions (ECBFs) in Nguyen and Sreenath (2016), which enforce forward invariance of set-based tasks with a higher relative degree.

The main contribution of this paper is a novel dynamic task-priority framework for an arbitrary number of equality- and set-based control tasks encoded by CLFs and CBFs, where equality and set-based tasks are control objectives that should be driven to a desired value and kept within a desired set, respectively. The framework builds on the CLF-based QP proposed for two equality tasks in Ames and Powell (2013) by extending it to an arbitrary number of equality tasks, unifying CLFs with CBFs via QPs to support set-based tasks as done in Ames et al. (2014b, 2017), and establishing any number of priority levels through a hierarchy of QPs. An important feature of this approach is that it yields strict priority between tasks at different priority levels, in the sense that tasks at lower-priority levels have no effect on the execution of tasks at higher-priority levels. The inclusion of set-based tasks at the dynamic level is a key novelty within task-priority control, which has to the best of our knowledge only been accounted for in Mansard et al. (2009).

The proposed scheme represents a holistic control approach since the QPs can be formulated in terms of the actuator inputs, instead of the commanded forces and torques. Consequently, the proposed framework also solves the control allocation problem. For task-priority control of robotic systems where computation of the actuator inputs from the commanded forces and torques is non-trivial, the unification of redundancy resolution and control allocation is a key advantage because strict priority between tasks can be ensured at all times. In contrast, redundancy resolution schemes that decouple dynamic control and control allocation, such as kinematic or operational space control, provide no a priori guarantee that the commanded forces and torques computed by the dynamic controller can be exactly allocated, since the commanded forces and torques are typically computed with no regard to physical actuator limits, rate constraints, or singularities of the actuator configuration matrix. If the commanded forces and torques cannot be exactly allocated, the forces and torques are usually allocated to actuator inputs by minimizing the allocation error (Johansen and Fossen, 2013), which is performed independently of the redundancy resolution algorithm. As a result, strict priority is lost and tasks become coupled whenever exact allocation is infeasible.

This paper is organized as follows. Section 2 presents background material related to CLFs and CBFs, before Section 3 introduces the proposed task-priority framework. Section 4 presents simulation results of the framework implemented on an articulated intervention autonomous underwater vehicle (AIAUV), while conclusions and future work can be found in Section 5.

2. BACKGROUND MATERIAL

In this section, the necessary background material will be presented. For compactness, we will slightly abuse notation and denote

$$L_g h(x) = \frac{\partial h(x)}{\partial x} g(x), \tag{1}$$

whenever $h(x)$ is a scalar or vector-valued function, and $g(x)$ is a vector field or a matrix. Note that (1) is only equal to the Lie derivative of $h(x)$ along $g(x)$ when $h(x)$ is a multivariable scalar function and $g(x)$ a vector field.

2.1 Model

Consider the nonlinear affine control system

$$\dot{x} = f(x) + g(x)u, \tag{2}$$

where f and g are locally Lipschitz, $x \in D \subset \mathbb{R}^l$ and $u \in U \subset \mathbb{R}^p$ is the set admissible control inputs. Let the locally Lipschitz vector-valued function $y = \sigma(x) - \sigma_d$ describe the error coordinates of some equality task $\sigma : \mathbb{R}^l \to \mathbb{R}^m$. Under the following assumption

$$L_g L_f^k y = 0, \quad 0 \leq k \leq \rho - 2 \tag{3}$$

$$L_g L_f^{\rho-1} y \neq 0, \tag{4}$$

the input-output dynamics becomes

$$y^{(\rho)} = \underbrace{L_f^{\rho} y(x)}_{b(x)} + \underbrace{L_g L_f^{\rho-1} y(x)}_{A(x)} u. \tag{5}$$

The system (2) can be decomposed into transverse dynamics states $\eta = \operatorname{col}\left(y, \dot{y}, \dots, y^{(\rho-1)}\right) \in X \subset \mathbb{R}^{\rho m}$ and internal dynamics states $z \in Z \subset \mathbb{R}^{l-\rho m}$ as follows

$$\dot{\eta} = \bar{f}(\eta, z) + \bar{g}(\eta, z)u, \tag{6a}$$

$$\dot{z} = f_z(\eta, z), \tag{6b}$$

with $\bar{f}(\eta, z) = F\eta + Gb(x)$ and $\bar{g}(\eta, z) = GA(x)$ where

$$F = \begin{bmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & I \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ I \end{bmatrix}, \tag{7}$$

where 0 is the $m \times m$ matrix of zeros and I is the $m \times m$ identity matrix.

2.2 Control Lyapunov Functions

A control Lyapunov function is a candidate Lyapunov function V, for which $\dot{V}$ can be made negative by appropriate selection of the control input u. In order to explicitly control the rate of exponential convergence, a specific type of CLF is defined in Ames et al. (2014a) as follows:

Definition 1. A continuously differentiable and positive definite function $V_\epsilon : X \to \mathbb{R}$ is said to be a rapidly exponentially stabilizing control Lyapunov function (RES-CLF) for the system (6) if there exists constants $c_1, c_2, c_3 > 0$ such that for all $0 < \epsilon < 1$ and for all states $(\eta, z) \in X \times Z$ it holds that

$$c_1 \|\eta\|^2 \leq V_\epsilon(\eta) \leq \frac{c_2}{\epsilon^2} \|\eta\|^2, \tag{8}$$

$$\inf_{u \in U} \left[L_{\bar{f}} V_\epsilon(\eta, z) + L_{\bar{g}} V_\epsilon(\eta, z)u + \frac{c_3}{\epsilon} V_\epsilon(\eta) \right] \leq 0. \tag{9}$$

Such a function can be constructed by solving the continuous time algebraic Riccati equation

$$F^T P + PF - PGG^T P + Q = 0, \tag{10}$$

for $P = P^T > 0$, where Q is any positive definite matrix. In order to stabilize the transverse dynamics at a rate ϵ define

$$V_\epsilon(\eta) = \eta^T \begin{bmatrix} \frac{1}{\epsilon} I & 0 \\ 0 & I \end{bmatrix} P \begin{bmatrix} \frac{1}{\epsilon} I & 0 \\ 0 & I \end{bmatrix} \eta := \eta^T P_\epsilon \eta. \tag{11}$$

When $A(x)$ has linearly independent rows, it can be shown that the time derivative of (11) satisfies (Ames et al., 2014a)

$$\inf_{u \in U} \left[L_{\bar{f}} V_\epsilon(\eta, z) + L_{\bar{g}} V_\epsilon(\eta, z)u \right] \leq -\frac{\gamma}{\epsilon} V_\epsilon(\eta), \tag{12}$$

where $\gamma := \frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)} > 0$ and

$$L_{\bar{f}} V_\epsilon(\eta, z) = \eta^T \left(F^T P_\epsilon + P_\epsilon F\right) \eta + 2\eta^T PGb, \tag{13}$$

$$L_{\bar{g}} V_\epsilon(\eta, z) = 2\eta^T P_\epsilon GA. \tag{14}$$

2.3 Control Barrier Functions

Control objectives described by inequalities or sets can be enforced by rendering the superlevel set

$$\mathcal{C} = \left\{ x \in D \subset \mathbb{R}^l : h(x) \geq 0 \right\}, \tag{15}$$

of some continuously differentiable function $h : D \to \mathbb{R}$ forward invariant (Ames et al., 2019).

Definition 2. Let $\mathcal{C} \subset D \subset \mathbb{R}^l$ be the superlevel set of a continuously differentiable function $h : D \to \mathbb{R}$, then h is a control barrier function (CBF) for the system (2) if there exists an extended class $\mathcal{K}_\infty$ function α such that

$$\sup_{u \in U} \left[L_f h(x) + L_g h(x) u \right] \geq -\alpha\left(h(x)\right), \tag{16}$$

for all $x \in D$.

The existence of a CBF implies that the superlevel set $\mathcal{C}$ of the function h can be rendered forward invariant by appropriate selection of the control input (Ames et al., 2017), which means that if $x(t_0) = x_0 \in \mathcal{C}$, then $x = x(t) \in \mathcal{C}$ for all $t \geq t_0$. Equivalently, if $h(x_0) \geq 0$, then $h(x) \geq 0$ for all $t \geq t_0$.

2.4 Exponential Control Barrier Functions

Definition 2 assumes that the relative degree of h is equal to one. However, safety-related tasks for robotic systems are often a function of the configuration variables only, meaning that they have a higher relative degree. Introduced in Nguyen and Sreenath (2016) and refined in Ames et al. (2019), exponential control barrier functions generalizes CBFs to functions $h(x)$ with arbitrary relative degree $r \geq 1$. To this end, we define $\eta_b = \operatorname{col}\Big(h(x), L_f h(x), L_f^2 h(x), \dots, L_f^{r-1} h(x)\Big)$ and assume that u can be chosen such that $L_f^r h(x) + L_g L_f^{r-1} h(x) u = \mu$ for a given $\mu \in U_\mu \subset \mathbb{R}$. The dynamics of $h(x)$ can then be written as a linear system

$$\dot{\eta}_b(x) = F\eta_b(x) + G\mu, \tag{17}$$

$$h(x) = C\eta_b(x), \tag{18}$$

where

$$F_b = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad G_b = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \tag{19}$$

$$C_b = \left[1 \ 0 \ \cdots \ 0\right]. \tag{20}$$

Choosing the state feedback $\mu = -K_\alpha \eta_b(x)$ results in $h(x(t)) = C_b e^{(F_b - G_b K_\alpha)t} \eta_b(x_0)$, which by the comparison lemma implies that if $\mu \geq -K_\alpha \eta_b(x)$ then $h(x(t)) \geq C_b e^{(F_b - G_b K_\alpha)t} \eta_b(x_0)$.

Definition 3. Given a set $\mathcal{C} \subset D \subset \mathbb{R}^l$ defined as the superlevel set of an r-times continuously differentiable function $h : D \to \mathbb{R}$, then h is an exponential control barrier function (ECBF) for the control system (2) if there exists a row vector $K_\alpha \in \mathbb{R}^r$ such that

$$\sup_{u \in U} \left[L_f^r h(x) + L_g L_f^{r-1} h(x) u \right] \geq -K_\alpha \eta_b(x), \tag{21}$$

$\forall\, x \in \operatorname{Int}(\mathcal{C})$ results in $h(x(t)) \geq C_b e^{(F_b - G_b K_\alpha)t} \eta_b(x_0) \geq 0$ whenever $h(x_0) \geq 0$.

2.5 Combining CLFs and ECBFs

The RES-CLF and ECBF conditions in (9) and (21) are both affine in the control input u, which means that the control problem can be formulated as a convex optimization problem, enabling the incorporation of control input saturation limits and rate constraints (Galloway et al., 2015). By employing RES-CLFs, the CLF-ECBF-based QP from (Ames et al., 2017; Ames et al., 2019) becomes:

$$\begin{aligned}
&\underset{u\in\mathbb{R}^m,\delta\in\mathbb{R}}{\text{minimize}} \quad \frac{1}{2}u^T H(x)u + c^T(x)u + w\delta^2 \\
&\text{subject to} \\
&L_{\bar f}V_\epsilon(\eta,z) + L_{\bar g}V_\epsilon(\eta,z)u \le -\frac{\gamma}{\epsilon}V_\epsilon + \delta, \\
&L_f^r h(x) + L_g L_f^{r-1}h(x)u \ge -K_\alpha \eta_b(x),
\end{aligned} \tag{22}$$

where $H\colon D \to \mathbb{R}^{m\times m}$ is any positive semi-definite matrix, $c\colon D \to \mathbb{R}^m$, and $\delta \in \mathbb{R}$ is a slack variable penalized by $w > 0$, ensuring the feasibility of the QP in case of conflicting set-based and equality-based control objectives.

3. QUADRATIC PROGRAMS FOR N EQUALITY- AND M SET-BASED CONTROL TASKS

This section extends the CLF-ECBF QP controller in (22) to an arbitrary number of equality- and set-based control tasks distributed to an arbitrary number of priority levels.

3.1 CLF Penalty Parameters as a Priority Measure

Inspired by Ames and Powell (2013), the QP in (22) can be extended to N equality-based control objectives by deriving the input-output dynamics for each control objective, i.e.

$$y_i^{(\rho_i)}(x) = \underbrace{L_f^{\rho_i}y_i(x)}_{b_i(x)} + \underbrace{L_g L_f^{\rho_i-1}y_i(x)}_{A_i(x)}u, \tag{23}$$

for each $i = 1,\dots,N$. Transverse dynamics states $\eta_i = \operatorname{col}\left(y_i, \dot y_i, \dots, y_i^{(\rho_i-1)}\right)$ and RES-CLFs $V_{\epsilon,i}$ can then be defined analogously to (6a), (7) and (11). Moreover, M set-based tasks described by the superlevel set $\mathcal{C}_j$ of some r_j times continuously differentiable function $h_j(x)$ can be included at the highest priority level (which is implied by no slack variables). The control input can then be obtained from the QP:

$$\begin{aligned}
&\underset{(u,\delta)\in\mathbb{R}^{m+N}}{\text{minimize}} \quad u^T H(x)u + c^T(x)u + \delta^T W\delta \\
&\text{subject to} \\
&L_{\bar f_i}V_{\epsilon,i} + L_{\bar g_i}V_{\epsilon,i}u \le -\frac{\gamma_i}{\epsilon}V_{\epsilon,i} + \delta_i, \quad i = 1,\dots,N, \\
&L_f^{r_k}h_k + L_g L_f^{r_k-1}h_k u \ge -K_{\alpha,k}\eta_{b,k}, \quad k = 1,\dots,M,
\end{aligned} \tag{24}$$

where $W \in \mathbb{R}^{N\times N}$ is a diagonal matrix of penalty parameters, $\eta_{b,k} = \operatorname{col}\left(h_k(x), L_f h_k(x), \dots, L_f^{r-1}h_k(x)\right)$ and

$$L_{\bar f_i}V_{\epsilon,i} = \eta_i^T\left(F_i^T P_{\epsilon,i} + P_{\epsilon,i}F_i\right)\eta_i + 2\eta_i^T P_i G_i b_i, \tag{25}$$

$$L_{\bar g_i}V_{\epsilon,i} = 2\eta_i^T P_{\epsilon,i}G_i A_i. \tag{26}$$

The equality tasks encoded by RES-CLFs are prioritized by adjusting the elements of the diagonal penalty matrix W. The satisfaction of all equality tasks are therefore described by a single objective function through the value of the slack variables δ and the penalty parameters in W. Whenever equality tasks are incompatible, this fact invariably leads to trade-off configurations that do not satisfy any of the tasks. Hence, strict priority between tasks cannot be achieved in the sense that lower-priority tasks have no effect on the execution of higher-priority tasks. As a result, it is challenging to include lower-priority optimization-based tasks since they will interfere with more critical higher-priority tasks such as end-effector control whenever the tasks are incompatible.

3.2 Main Result: Enforcing Strict Priority Between a Selection of Tasks

In order to establish more than two strict priority levels, we propose to solve a quadratic program for every priority level as suggested for kinematic control in Kanoun et al. (2011). The idea is to begin by computing a control input according to (24) that only accounts for safety-related set-based tasks and equality tasks at the highest priority level. Subsequently, a new quadratic program is solved for each priority level, refining the previous solution in an attempt to satisfy lower-priority tasks without affecting the execution of higher-priority tasks.

Consider N equality tasks and M set-based tasks distributed to k priority levels, with $N = N_1 + \ldots + N_k$ and $M = M_1 + \ldots + M_k$, where N_i and M_i denotes the number of equality and set-based tasks at priority level i, respectively. A control input u_1^* that disregards all lower-priority tasks is obtained by solving (24) with $i = 1,\dots,N_1$ and $k = 1,\dots,M_1$. If the system is redundant with respect to these $N_1 + M_1$ tasks, the control input u_1^* can be refined without affecting how the N_1 higher-priority equality tasks are executed by enforcing

$$L_{\bar f_i}V_{\epsilon,i} + L_{\bar g_i}V_{\epsilon,i}u \le L_{\bar f_i}V_{\epsilon,i} + L_{\bar g_i}V_{\epsilon,i}u_1^* \tag{27}$$

which implies that $L_{\bar g_i}V_{\epsilon,i}u \le L_{\bar g_i}V_{\epsilon,i}u_1^*$ for all $i = 1,\dots,N_1$. Similarly, the higher-priority set-based tasks are unaffected by enforcing

$$L_g L_f^{r_k-1}h_k u \ge L_g L_f^{r_k-1}h_k u_1^*, \tag{28}$$

for all $k = 1,\dots,M_1$. Consider N_2 additional equality-based tasks and M_2 additional set-based tasks. The control input u_1^* can be modified to account for lower-priority tasks without affecting how the N_1 and M_1 equality- and set-based tasks are executed by solving:

$$\begin{aligned}
&\underset{(u,\delta,s)\in\mathbb{R}^{m+N_2+M_2}}{\text{minimize}} \quad u^T H(x)u + c^T(x)u + \delta^T W_2\delta + s^T K_2 s \\
&\text{subject to} \\
&L_{\bar g_i}V_{\epsilon,i}u \le L_{\bar g_i}V_{\epsilon,i}u_1^*, && {\scriptstyle i=1,\dots,N_1,} \\
&L_{\bar f_j}V_{\epsilon,j} + L_{\bar g_j}V_{\epsilon,j}u \le -\frac{\gamma_j}{\epsilon}V_{\epsilon,j} + \delta_j, && {\scriptstyle j=N_1+1,\dots,N_1+N_2,} \\
&L_g L_f^{r_k-1}h_k u \ge L_g L_f^{r_k-1}h_k u_1^*, && {\scriptstyle k=1,\dots,M_1,} \\
&L_f^{r_l}h_l + L_g L_f^{r_l-1}h_l u \ge -K_{\alpha,l}\eta_{b,l} - s_l, && {\scriptstyle l=M_1+1,\dots,M_1+M_2,}
\end{aligned} \tag{29}$$

where slack variables s penalized by the elements in the diagonal matrix $K > 0$ have been added to the lower-priority set-based tasks enforced through ECBFs to ensure feasibility of the optimization problem.

By observing that the solution u_2^* to (29) enforces the constraints $L_{\bar g_i}V_{\epsilon,i}u_2^* \le L_{\bar g_i}V_{\epsilon,i}u_1^*$ and $L_g L_f^{r_k-1}h_k u_2^* \ge L_g L_f^{r_k-1}h_k u_1^*$ for all i and k, it is straightforward to generalize (29) to an arbitrary priority level n:

$$\begin{aligned}
&\underset{(u,\delta,s)\in\mathbb{R}^{m+N_n+M_n}}{\text{minimize}} \quad u^T H u + c^T u + \delta^T W_n \delta + s^T K_n s \\
&\quad\text{subject to} \\
&L_{\bar{g}_i} V_{\epsilon,i} u \le L_{\bar{g}_i} V_{\epsilon,i} u^*_{n-1}, && {\scriptstyle i=1,\dots,\bar{N}_{n-1},} \\
&L_{\bar{f}_j} V_{\epsilon,j} + L_{\bar{g}_j} V_{\epsilon,j} u \le -\frac{\gamma_j}{\epsilon} V_{\epsilon,j} + \delta_j, && {\scriptstyle j=\bar{N}_{n-1}+1,\dots,\bar{N}_n,} \\
&L_g L_f^{r_k-1} h_k u \ge L_g L_f^{r_k-1} h_k u^*_{n-1}, && {\scriptstyle k=1,\dots,\bar{M}_{n-1},} \\
&L_f^{r_l} h_l + L_g L_f^{r_l-1} h_l u \ge -K_{\alpha,l}\eta_{b,l} - s_l, && {\scriptstyle l=\bar{M}_{n-1}+1,\dots,\bar{M}_n,}
\end{aligned} \tag{30}$$

where $\bar{N}_n = N_1 + N_2 + \ldots + N_n$ and $\bar{M}_n = M_1 + M_2 + \ldots + M_n$. Note that the objective function is slightly different at every priority level, since the slack variables δ and s always correspond to tasks at the current priority level. This prevents trade-off configurations where none of the tasks are satisfied from occurring. The procedure is summarized in Algorithm 1.

Algorithm 1 Task priority CLF-ECBF QP controller

```
Input: $H(x)$, $c(x)$, $V_{\epsilon,i}(\eta_i)$, $i = 1,\dots,N$, $h_j(x)$, $j = 1,\dots,M$.
Output: $u$
1: Solve (24) to obtain $u_1^*$ with $i = 1,\dots,N_1$, $k = 1,\dots,M_1$.
2: for $p = 2$ to $k$ do
3:     Solve (30) to obtain $u_p^*$.
4: end for
5: return $u = u_k^*$.
```

4. SIMULATIONS

In this section, the proposed hierarchical control scheme is validated in simulation on an AIAUV based on the Eelume robot (Schmidt-Didlaukies et al., 2018; Liljebäck and Mills, 2017) depicted in Fig. 1. The AIAUV is a floating base manipulator, with $n+1$ links interconnected by n joints, where link 1 is the tail, or base link and link $n+1$ is the head. The simulation model has $n = 8$ single DOF and revolute joints and $p = 7$ thrusters. The system configuration is described by $\xi = \operatorname{col}(p^i_{ib}, q, \theta) \in \mathbb{R}^{7+n}$, where $p^i_{ib} \in \mathbb{R}^3$ is the position of the base of the AIAUV in an inertial frame, $q = \operatorname{col}(\eta, \epsilon) \in \mathbb{R}^4$ is a unit quaternion describing the orientation of the base and $\theta = \operatorname{col}(\theta_1, \dots, \theta_n) \in \mathbb{R}^n$ are the joint angles. The joint velocities are given by $\dot{\theta}$ and the linear and angular velocities of the base frame with respect to an inertial frame are denoted v^b_{ib} and ω^b_{ib}, respectively. These velocities are collected in the velocity vector $\zeta = \operatorname{col}(v^b_{ib}, \omega^b_{ib}, \dot{\theta}) \in \mathbb{R}^{6+n}$. The equations of motion are given by (Schmidt-Didlaukies et al., 2018)

$$\dot{\xi} = J_\xi(q)\zeta, \tag{31}$$

$$M(\theta)\dot{\zeta} + C(\theta,\zeta)\zeta + D(\theta,\zeta)\zeta + g(\xi) = B(\theta)u, \tag{32}$$

where $M(\theta)$ is the inertia matrix including hydrodynamic added mass, $C(\theta,\zeta)$ is the Coriolis-centripetal matrix including hydrodynamic added mass, $D(\theta,\zeta)$ is the damping matrix, $g(\xi)$ is the vector of gravitational and buoyancy forces and moments, $B(\theta)$ is the actuator configuration matrix and $u = \operatorname{col}(u_t, u_j) \in \mathbb{R}^{p+n}$ consists of the thruster inputs $u_t \in \mathbb{R}^p$ and joint torque inputs $u_j \in \mathbb{R}^n$. Moreover, the kinematic transformation matrix is given by

$$J_\xi(q) = \begin{bmatrix} R^i_b(q) & 0_{3\times3} & 0_{3\times n} \\ 0_{4\times3} & T_q(q) & 0_{4\times n} \\ 0_{n\times3} & 0_{n\times3} & I_n \end{bmatrix}, T_q(q) = \frac{1}{2}\begin{bmatrix} -\epsilon^T \\ \eta I_3 + [\epsilon]_\times \end{bmatrix}, \tag{33}$$

where $R^i_b(q) \in SO(3)$ is a rotation matrix describing the rotation between the base and inertial frame and $[\cdot]_\times : \mathbb{R}^3 \to \mathfrak{so}(3) \subset \mathbb{R}^{3\times3}$ denotes the skew symmetric map.

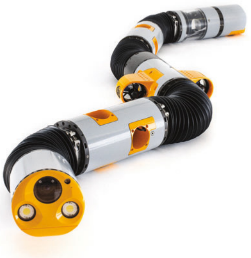

Fig. 1. The Eelume AIAUV (Courtesy of Eelume)

By defining $x = \operatorname{col}(x_1, x_2) = \operatorname{col}(\xi, \zeta)$, the equations of motion can be rewritten in state space form

$$\dot{x} = f(x) + g(x)u, \tag{34}$$

where

$$f(x) = \begin{bmatrix} J_\xi(x_1)x_2 \\ -M(x_1)^{-1}\left(C(x)x_2 + D(x)x_2 + g(x_1)\right) \end{bmatrix}, \tag{35}$$

$$g(x) = \begin{bmatrix} 0 \\ M(x_1)^{-1}B(x_1) \end{bmatrix}. \tag{36}$$

With $6 + n = 14$ DOFs and $p + n = 15$ control inputs, the system is overactuated, since the number of actuators is greater than the number of DOFs. Moreover, 14 DOFs imply that the system is redundant with respect to typical tasks such as end-effector configuration control.

We consider four equality-based tasks and three set-based tasks, at three different priority levels. The set-based tasks are safety-related and are thus placed at the highest priority level. The safety-related tasks consist of end-effector collision avoidance, actuator configuration matrix singularity avoidance and joint limit avoidance. The purpose of the end-effector collision avoidance task is to avoid a spherical obstacle with radius $r_{\text{obs}} \in \mathbb{R}$. To this end, the scalar distance measure between the center of the obstacle and the end-effector is employed as a set-based task $\sigma_a \in \mathbb{R}$. In order to ensure that the distance from the end-effector to the center of the spherical obstacle is always greater than some lower limit, we enforce the positivity of the following ECBF

$$h_1 = \underbrace{\sqrt{\left(p^i_{\text{obs}} - p^i_{ie}\right)^T \left(p^i_{\text{obs}} - p^i_{ie}\right)}}_{\sigma_a} - (r_{\text{obs}} + \epsilon), \tag{37}$$

where $\epsilon \in \mathbb{R}$ defines an inaccessible safety region around the spherical obstacle and p^i_{ie} and p^i_{obs} are the positions of the end-effector and the center of the spherical obstacle in an inertial frame, respectively.

Rank deficiency of the actuator configuration matrix $B(\theta)$ was pointed out in Sverdrup-Thygeson et al. (2018), and implies that no force or moment can be generated in certain directions in the vector space $\mathbb{R}^{6+n}$ belonging to τ. Inspired by the manipulability measure (Yoshikawa, 1985), the actuation measure $\sigma_b = \det\left(B(\theta)B^T(\theta)\right)$ is introduced as a high-priority set-based task to prevent singular configurations. The actuation measure is kept above a minimum value $\sigma_{b,\min}$ through the following ECBF

$$h_2 = \det\left(B(\theta)B^T(\theta)\right) - \sigma_{b,\min}. \tag{38}$$

The third safety-related task is the joint limit avoidance task $\sigma_c = \theta \in \mathbb{R}^n$, which has both lower and upper limits.

Hence, $2n$ ECBFs are needed of the form

$$h_{j+2} = \theta_j - \theta_{j,\min}, \tag{39}$$

$$h_{j+2+n} = \theta_{j,\max} - \theta_j. \tag{40}$$

for $j = 1, \ldots, n$.

The second priority level contains the equality-based end-effector positioning and orientation tasks

$$y_1 = p^i_{ie} - p^i_{d,e}, \tag{41}$$

$$y_2 = \tilde{\epsilon}, \tag{42}$$

where $p^i_{d,e}$ is the desired end-effector position and $\tilde{\epsilon}$ is the imaginary part of the quaternion error vector $\tilde{q} = q_d \otimes q^*$, which is given by

$$\tilde{\epsilon} = \eta\epsilon_d - \eta_d\epsilon + [\epsilon]_\times \epsilon_d, \tag{43}$$

where q_d and q are the quaternion representations of the desired and measured orientation of the end-effector, respectively.

In order to minimize movement of the base while repositioning the end-effector, a base positioning task is defined at the third priority level

$$y_3 = p^i_{ib} - p^i_{d,b}, \tag{44}$$

where $p^i_{ib} \in \mathbb{R}^3$ and $p^i_{d,b} \in \mathbb{R}^3$ are the measured and desired positions of the AIAUV base in the inertial frame, respectively. Note that the end-effector positioning and orientation and base positioning tasks only consume 9 DOFs, which entails that there are still 5 uncontrolled DOFs if all set-based tasks are inactive. Stability of the entire system can therefore only be guaranteed if the resulting zero dynamics is asymptotically stable. Instead of performing a complicated analysis of the zero dynamics, a joint velocity regulation task is designed to eliminate the residual DOFs of the system $y_4 = \dot{\theta}$, where $\dot{\theta} \in \mathbb{R}^n$ is the vector of joint velocities.

The input-output dynamics of the equality tasks are then obtained from (23) such that transverse dynamics states η_i and RES-CLFs $V_{\epsilon,i}$ can be defined analogously to (6a), (7) and (11), with $\rho_1 = \rho_2 = \rho_3 = 2$ and $\rho_4 = 1$, where ρ_i denotes the amount of times y_i has to be differentiated for the input to appear. Furthermore, the set-based tasks in (37)-(40) all have to be differentiated twice with respect to time for the input to show up, hence $r_1 = \cdots = r_{18} = 2$.

The design matrix $H(x)$ and design vector $c(x)$ in the objective functions are selected by minimizing the virtual control input $\mu = Au + b$ quadratically as done in Ames and Powell (2013); Galloway et al. (2015), where

$$A(x) = \begin{bmatrix} A_1(x) \\ A_2(x) \\ A_3(x) \\ A_4(x) \end{bmatrix}, \quad b(x) = \begin{bmatrix} b_1(x) \\ b_2(x) \\ b_3(x) \\ b_4(x) \end{bmatrix}. \tag{45}$$

In terms of u, this yields

$$\mu^T\mu = u^T A^T A u + 2b^T A u + b^T b, \tag{46}$$

which implies that $H(x) = A^T(x)A(x)$ and $c^T(x) = 2b^T(x)A(x)$.

Table 1. Equality task convergence rates ϵ and penalty parameters w

	y_1	y_2	y_3	y_4
ϵ	1.2	0.2	1.2	0.5
w	60	60	10	10

According to Algorithm 1, we solve the following QP:

$$\begin{aligned} \underset{u \in \mathbb{R}^{15}, (\delta_1,\delta_2) \in \mathbb{R}^2}{\text{minimize}} \quad & u^T A^T A u + 2b^T A u + w_1\delta_1^2 + w_2\delta_2^2 \\ \text{subject to} \quad & \\ & L_{\bar{f}_i} V_{\epsilon,i} + L_{\bar{g}_i} V_{\epsilon,i} u \le -\frac{\gamma_1}{\epsilon_i} V_{\epsilon,i} + \delta_i, \quad i = 1, 2, \\ & L_f^2 h_k + L_g L_f h_k u \ge -K_{\alpha,k}\eta_{b,k}, \qquad k = 1, \ldots, 18, \\ & -u_{max} \le u \le u_{max}, \\ & -\Delta u_{max} \le \Delta u \le \Delta u_{max}, \end{aligned} \tag{47}$$

where $\Delta u = u - u_{\text{prev}}$ is the change in control input, u_{prev} is the control input at the last sample, and $u_{max} = \text{col}\,(50, \ldots, 50)$ and $\Delta u_{max} = \text{col}\,(0.1, \ldots, 0.1)$ are thruster and joint torque limits and rate constraints, respectively. The QP in (47) yields a control input $u = u_1^*$ that only accounts for the safety-related tasks and the end-effector positioning and orientation tasks. The solution u_1^* is refined by utilizing the excess DOFs of the system in an attempt to keep the base stationary and minimize the joint velocities through the QP:

$$\begin{aligned} \underset{u \in \mathbb{R}^{15}, (\delta_3,\delta_4) \in \mathbb{R}^2}{\text{minimize}} \quad & u^T A^T A u + 2b^T A u + w_3\delta_3^2 + w_4\delta_4^2 \\ \text{subject to} \quad & \\ & L_{\bar{g}_i} V_{\epsilon,i} u \le L_{\bar{g}_i} V_{\epsilon,i} u_i^*, \qquad i = 1, 2, \\ & L_{\bar{f}_j} V_{\epsilon,j} + L_{\bar{g}_j} V_{\epsilon,j} u \le -\frac{\gamma_j}{\epsilon_j} V_{\epsilon,j} + \delta_j, \quad j = 3, 4, \\ & L_g L_f h_k u \ge L_g L_f h_k u_1^*, \qquad k = 1, \ldots, 18, \\ & -u_{max} \le u \le u_{max}, \\ & -\Delta u_{max} \le \Delta u \le \Delta u_{max}, \end{aligned} \tag{48}$$

which yields the final control input $u = u_2^*$ that is applied to the AIAUV. The equality task control parameters are listed in Table 1, while $K_{\alpha,k} = [3, 4]$ for all $k = 1, \ldots, 18$.

We remark that the optimization problems are formulated in terms of the thruster and joint torque control inputs u, and not the commanded forces and torques $\tau = Bu$. Consequently, the proposed framework also solves the control allocation problem, which had to be solved separately in previous works (Sverdrup-Thygeson et al., 2018; Borlaug et al., 2019). By unifying redundancy resolution, dynamic control and control allocation, strict priority among tasks can always be ensured. The same guarantee does not hold for redundancy resolution schemes that decouple dynamic control and control allocation, since the commanded forces and torques may not be exactly allocable, leading to a loss of priority among tasks.

Simulations were performed in Matlab/Simulink using the ode3 solver with a fixed step-size of 0.01. Simulation results are presented in Figs. 2 to 5. From Figs. 4 and 5 we observe that the high-priority set-based tasks are satisfied at all times. In general, the redundancy of the system is exploited such that the lower-priority equality tasks are satisfied even when higher-priority set-based tasks are at their limits and consuming DOFs. For instance, the actuation measure is kept above a minimum value of 0.1, which avoids singular configurations of the actuation configuration

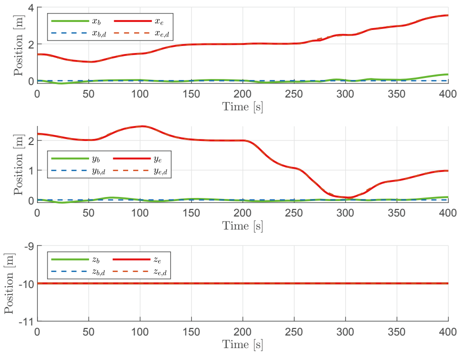

Fig. 2. The position of the end-effector p^i_{ie} and base p^i_{ib}.

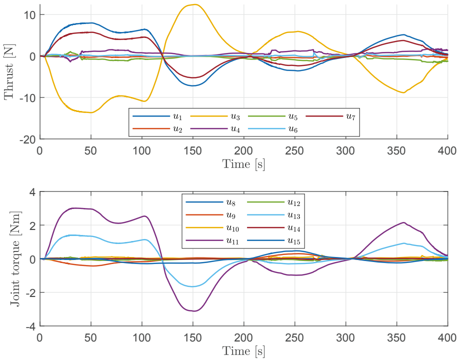

Fig. 3. The thruster and joint torque control inputs.

matrix from occurring and thereby reducing the magnitude and/or rates of change of the control inputs, at the cost of maneuverability (Johansen et al., 2004). However, the high-priority collision avoidance task results in a small deviation in the lower-priority end-effector positioning task. Specifically, we observe from Fig. 2 and Fig. 4 that the x-coordinate of the end-effector position deviates slightly from its reference and that the distance to the center of the spherical obstacle is at its minimum value between $t \simeq 268$ s and $t \simeq 281$ s.

After $t \geq 350$ s, the end-effector position is commanded outside of the manipulator workspace (when the base is kept at its current position), which implies that the lower-priority base positioning task is no longer compatible with the higher-priority end-effector positioning task. As desired, the strict priority between tasks is kept at all time, such that the end-effector position converges to its desired value at the expense of a greater error in the base position.

Finally, we note from Fig. 3 that the thruster and joint torque control inputs are smooth and well within the physical limitations of the Eelume robot.

5. CONCLUSIONS AND FUTURE WORK

This paper has presented a novel task-priority framework for redundancy resolution, dynamic control and control allocation of redundant robotic systems based on a hierarchy of CLF- and CBF-based QPs. The framework provides strict priority, ensuring that lower-priority tasks have no effect on higher-priority tasks, by solving additional QPs to establish distinct priority levels. As a result, lower-priority control objectives can be safely included, without affecting the execution of higher-priority mission-related or safety-related tasks. Additionally, a soft priority measure in the form of slack variables can be utilized in order to prioritize tasks at the same priority level, resulting in considerable design freedom.

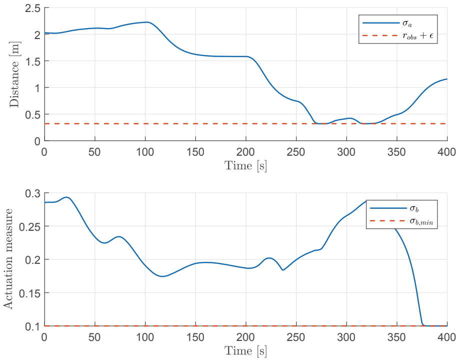

Fig. 4. The distance to the center of the spherical obstacle σ_a and its minimum value $r_{\text{obs}} + \epsilon$, and the actuation measure σ_b and its minimum value $\sigma_{b,\min}$.

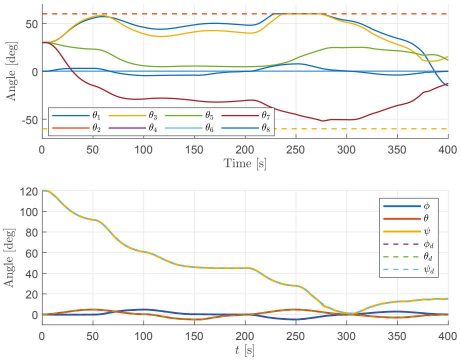

Fig. 5. The joint angles θ, their maximum and minimum limits $\theta_{\max} = 60°$ and $\theta_{\min} = -60°$, and the orientation of the end-effector, represented by the roll-pitch-yaw Euler angles ϕ, θ and ψ.

The proposed framework has been verified in simulations for an AIAUV, which is an overactuated and redundant robotic system. For these types of systems, the proposed task-priority framework also solves the control allocation problem, which is highly advantageous since control input bounds and rate constraints can be accounted for when resolving redundancy, effectively avoiding a situation in which commanded generalized forces and torques cannot be allocated explicitly, leading to a loss of priority among tasks.

Future work is aimed at investigating the robustness of the proposed framework with respect to modeling inaccuracies. This is especially relevant for an underwater vehicle application such as an AIAUV, where accurate identification of the dynamic model parameters is difficult (Antonelli, 2018). An experimental implementation of the proposed control system on an AIAUV will further validate the framework.

REFERENCES

Ames, A.D., Galloway, K., Sreenath, K., and Grizzle, J.W. (2014a). Rapidly exponentially stabilizing control Lyapunov functions and hybrid zero dynamics. *IEEE Trans. Autom. Control*, 59(4), 876–891.

Ames, A.D., Grizzle, J.W., and Tabuada, P. (2014b). Control barrier function based quadratic programs with application to adaptive cruise control. In *Proc. 53rd*

IEEE Conf. Decision and Control. Los Angeles, CA, USA.

Ames, A.D., Xu, X., Grizzle, J.W., and Tabuada, P. (2017). Control barrier function based quadratic programs for safety critical systems. *IEEE Trans. Autom. Control*, 62(8), 3861–3876.

Ames, A. and Powell, M. (2013). Towards the unification of locomotion and manipulation through control Lyapunov functions and quadratic programs. *Control of Cyber-Physical Systems*, 219–240.

Ames, A.D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., and Tabuada, P. (2019). Control barrier functions: Theory and applications. In *Proc. 2019 European Control Conference*. Naples, Italy.

Antonelli, G. (2018). *Underwater Robots*. Springer, 4th edition.

Artstein, Z. (1983). Stabilization with relaxed controls. *Nonlinear Analysis: Theory, Methods & Applications*, 7(11), 1163–1173.

Borlaug, I.L.G., Pettersen, K.Y., and Gravdahl, J.T. (2019). Tracking control of an articulated intervention-auv in 6 DOF using the generalized super-twisting algorithm. In *Proc. 2019 American Control Conference*. Philadelphia, PA, USA.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Forsgren, A., Gill, P.E., and Wright, M.H. (2002). Interior methods for nonlinear optimization. *SIAM Review*, 44(4), 525–597.

Freeman, R.A. and Kokotović, P. (1996). *Robust Nonlinear Control Design*. Birkhäuser Boston.

Galloway, K., Sreenath, K., Ames, A.D., and Grizzle, J.W. (2015). Torque saturation in bipedal robotic walking through control Lyapunov function-based quadratic programs. *IEEE Access*, 3, 323–332.

Hanafusa, H., Yoshikawa, T., and Nakamura, Y. (1981). Analysis and control of articulated robot arms with redundancy. In *Proc. 8th IFAC World Congress*. Kyoto, Japan.

Johansen, T.A., Fossen, T.I., and Berge, S.P. (2004). Constrained nonlinear control allocation with singularity avoidance using sequential quadratic programming. *IEEE Trans. Control Syst. Technol*, 12(1), 211–216.

Johansen, T.A. and Fossen, T.I. (2013). Control allocation – a survey. *Automatica*, 49, 1087–1103.

Kanoun, O., Lamiraux, F., and Wieber, P. (2011). Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task. *IEEE Transactions on Robotics*, 27(4), 785–792.

Khatib, O. (1987). A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1), 43–53.

Liljebäck, P. and Mills, R. (2017). Eelume: A flexible and subsea resident IMR vehicle. In *OCEANS 2017*. Aberdeen, Scotland.

Mansard, N., Khatib, O., and Kheddar, A. (2009). A unified approach to integrate unilateral constraints in the stack of tasks. *IEEE Transactions on Robotics*, 25(3), 670–685.

Moe, S., Antonelli, G., Teel, A.R., Pettersen, K.Y., and Schrimpf, J. (2016). Set-based tasks within the singularity-robust multiple task-priority inverse kinematics framework: General formulation, stability analysis, and experimental results. *Frontiers in Robotics and AI*, 3, 16.

Nakamura, Y., Hanafusa, H., and Yoshikawa, T. (1987). Task-priority based redundancy control of robot manipulators. *The International Journal of Robotics Research*, 6(2), 3–15.

Nguyen, Q. and Sreenath, K. (2016). Exponential control barrier functions for enforcing high relative-degree safety-critical constraints. In *Proc. 2016 American Control Conf.* Boston, MA, USA.

Petersen, I.R. and Barmish, B.R. (1987). Control effort considerations in the stabilization of uncertain dynamical systems. *Systems & Control Letters*, 9(5), 417–422.

Prajna, S. (2006). Barrier certificates for nonlinear model validation. *Automatica*, 42(1), 117–126.

Prajna, S. and Jadbabaie, A. (2004). Safety verification of hybrid systems using barrier certificates. In *Hybrid Systems: Computation and Control*, 477–492. Springer.

Romdlony, M.Z. and Jayawardhana, B. (2014). Uniting control Lyapunov and control barrier functions. In *Proc. 53rd IEEE Conf. on Decision and Control*. Los Angeles, CA, USA.

Romdlony, M.Z. and Jayawardhana, B. (2016). Stabilization with guaranteed safety using control Lyapunov –barrier function. *Automatica*, 66, 39–47.

Schmidt-Didlaukies, H.M., Sørensen, A.J., and Pettersen, K.Y. (2018). Modeling of articulated underwater robots for simulation and control. In *Proc. 2018 IEEE OES AUV Symposium*. Porto, Portugal.

Sentis, L. and Khatib, O. (2004). Prioritized multi-objective dynamics and control of robots in human environments. In *Proc. 4th IEEE/RAS International Conference on Humanoid Robots, 2004*. Santa Monica, CA, USA.

Siciliano, B. and Slotine, J..E. (1991). A general framework for managing multiple tasks in highly redundant robotic systems. In *Proc. 5th Int. Conf. Advanced Robotics*. Pisa, Italy.

Simetti, E. and Casalino, G. (2016). A novel practical technique to integrate inequality control objectives and task transitions in priority based control. *Journal of Intelligent & Robotic Systems*, 84(1), 877–902.

Sontag, E.D. (1983). A Lyapunov-like characterization of asymptotic controllability. *SIAM Journal on Control and Optimization*, 21(3), 462–471.

Sontag, E.D. (1989). A universal construction of artstein's theorem on nonlinear stabilization. *Systems & Control Letters*, 13(2), 117–123.

Sverdrup-Thygeson, J., Kelasidi, E., Pettersen, K.Y., and Gravdahl, J.T. (2018). The Underwater Swimming Manipulator – A Bioinspired Solution for Subsea Operations. *IEEE Journal of Oceanic Engineering*, 43(2), 402–417.

Tee, K.P., Ge, S.S., and Tay, E.H. (2009). Barrier Lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*, 45(4), 918–927.

Wieland, P. and Allgöwer, F. (2007). Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12), 462–467.

Yoshikawa, T. (1985). Manipulability of robotic mechanisms. *The International Journal of Robotics Research*, 4(2), 3–9.