Robust Task-Priority Impedance Control for Vehicle-Manipulator Systems

Bjørn Kåre Sæbø, Kristin Y. Pettersen and Jan Tommy Gravdahl

***Abstract*— Enabling vehicle-manipulator systems to perform complex and precise intervention operations requires a robust control framework capable of handling redundancy and interaction forces. In addition, it is desirable that the method allows completion of several tasks with a strict prioritization to keep safety critical tasks unaffected by other goals. In this paper, a control method is developed for a broad class of vehicle-manipulator systems, with the main use case being floating base underwater vehicle manipulator systems subject to hydrodynamic forces. In underwater applications, large model uncertainties will be present due to hydrodynamic and hydrostatic effects, unknown disturbances and modelling errors. This means that a viable solution must be robust to disturbances and modelling errors, while also satisfying strict task priorities and compliant contact behaviour. To achieve this, a robust impedance based task-priority control method is presented, and its stability and robustness properties are proven. The method is validated in a simulation of an articulated intervention autonomous underwater vehicle (AIAUV).**

I. INTRODUCTION

The initial motivation for this paper came from the needs of underwater inspection, maintenance, and repair (IMR) tasks. Today, IMR tasks are performed using human divers or remotely operated vehicles (ROVs). ROVs are controlled by human operators, often situated on a supporting surface vessel to which the ROV is tethered. ROV missions are therefore costly, time-consuming, and have large environmental impacts. We have just entered the United Nations' ocean decade, and as the number of subsea operations increases, alternative solutions are required. One alternative is the use of autonomous underwater vehicles (AUVs) which can operate without human supervision. Typical AUVs are torpedo-shaped and designed for survey missions but lack the capability to interact with their environment and standing still in water (hovering). Thus, they can complete some inspection tasks, but not maintenance and repair. In this paper, we want to enable mobile, floating base robots that can not only "fly and see" untethered, but also perform complex tasks with precise manipulator forces. Such robots underwater are named intervention AUVs (I-AUVs), and constitute a subset of vehicle-manipulator systems (VMSs).

VMSs are inherently redundant in regard to the end-effector position, as there will always be the choice between moving the base and moving the manipulator arm to reach a certain configuration. A potential solution must therefore be able to deal with this redundancy. For intervention tasks, limited interaction forces are required to avoid damaging equipment or the vehicle. A much used solution for this is impedance control methods, which aim to control the apparent inertia, damping and stiffness experienced by the environment in interaction with a manipulator. In underwater applications, hydrodynamic and hydrostatic effects introduce significant disturbances and modelling errors. We will target the particular challenging case of underwater vehicle-manipulator systems (UVMSs), with the goal of providing a framework that can be generalized to other VMSs and to redundant fixed-base robot manipulators.

The redundancy of UVMSs allows the completion of several tasks at once, such as moving the end-effector while using redundant degrees of freedom (DoF) to avoid collisions or maintaining an energy efficient configuration. Using task-priority methods is therefore a beneficial way to perform redundancy resolution, and a structured approach to build up autonomous behavior. In safety-critical operations, strict prioritization of tasks is desired, to ensure that the highest prioritized safety related tasks are unaffected by other goals.

Task priority control methods typically use either null space operators [1] or optimization based approaches. Methods with null space operators include task-priority inverse kinematics (IK) [2], [3] and the operational space formulation (OSF) [4]. IK does redundancy resolution at the velocity level, building upon the fundamental assumption that the kinematics and dynamics can be decoupled, and hence it is not well suited for UVMSs for which the motion of the arm and the motion of the base are tightly connected. OSF makes the redundancy resolution at the acceleration level. Optimization based techniques such as [5], [6] allow the inclusion of set-based inequality constraints, but utilizes soft priorities, meaning that the highest prioritized tasks may be disturbed by those with lower priority.

Previous work on task priority control of UVMSs include

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, through the ERC Advanced Grant 101017697-CRÈME. The work is also supported by the Research Council of Norway through the Centres of Excellence funding scheme, project No. 223254 – NTNU AMOS.

The authors are in the Centre for Autonomous Marine Operations and Systems (NTNU AMOS), Department of Engineering Cybernetics, Norwegian University of Science and Technology, NTNU, NO-7491 Trondheim, Norway {bjorn.k.sabo, kristin.y.pettersen, jan.tommy.gravdahl}@ntnu.no

[7] where a task-priority IK approach is applied to allow *two* simultaneous tasks, and [8] which uses an optimization technique with control Lyapunov functions and control barrier function based QPs. The latter allows strict priority for different groups of tasks, and soft priorities for tasks in the same group. An approach with combined kinematic and dynamic control of UVMSs is proposed in [9], where a singularity-robust multiple task-priority framework is combined with a robust sliding mode controller for the dynamics and the stability is proven for the combined control system. The assumption of decoupled kinematics and dynamics is therefore not required. These methods do not allow the specification of interaction behavior required for our use case.

A general framework for impedance control was developed in [10]. The idea has been applied for interaction tasks with UVMSs previously, for example in [11], where sliding mode impedance control is applied for contact tasks with an I-AUV, and [12] which uses adaptive admittance control to achieve contact force tracking. Both of these approaches rely on measurements of the contact force through a sensor in the manipulator. A variant of impedance control where the natural inertia of the system is kept, and contact stiffness and damping can be specified is called compliance control[13]. This is easier to tune than impedance control. It is also more robust to modelling errors, which is beneficial for UVMSs due to the uncertainties in hydrodynamic effects.

Compliance control has been combined with task priority frameworks previously, as in [14] where a framework is developed for fixed base redundant manipulators, that ensures compliant behavior in the null space of the main task. In [15] a regulation algorithm is developed that dynamically decouples the tasks to allow a strict task priority framework with compliant behavior in the whole task space. This provides a framework with less reliance on accurate model knowledge than OSF, but has the downside of only achieving setpoint regulation, not trajectory tracking. This limitation is removed in [13], but with the new constraint that the task dimension must match the number of DoF of the system. In the more recent [16] and [17], a control law using a sliding variable is developed that allows both singular tasks, arbitrary large task-dimensions and tracking control. However, the control law is expressed in task velocity coordinates instead of position, giving a less intuitive specification of tasks. Additionally, the tuning of the desired compliance is coupled with the tuning of the sliding parameter. To enable tuning the sliding variable and compliance behavior separately, we base our approach on the work in [13].

In this paper, the control law from [13] is extended to the more general class of VMSs and modified to allow compensation of hydrodynamic effects. It is also extended with a sliding variable and sliding mode control to achieve improved convergence properties and robustness with a significantly simpler analysis. The main contributions of the work presented in this paper can thus be summarized as follows: A robust control law for VMSs is developed, which allows utilizing kinematic redundancy for the completion of simultaneous tasks while specifying compliant behavior during interaction. It is developed to address the challenges of floating-base underwater systems, but can also be applied to fixed-base or mobile systems on land. The closed-loop system is proven to be UAS, with ES dynamics for the highest prioritized task, as well as exponentially stable task dynamics for lower priority tasks when all higher priority tasks have converged. Finally, its effectiveness is verified in a simulation study of an AIAUV.

The paper is organized as follows. In Section II, the mathematical model of a general VMS systems is presented with some specific considerations for UVMS. In Section III, mathematical background is provided, and the task space is defined. The proposed control law is presented in Section IV, and its stability and robustness properties are proven in Section V. The control law is then validated through simulations in Section VI. Finally, conclusions and future work are presented in Section VII.

II. MATHEMATICAL MODEL

We consider general VMSs for which the equations of motion expressed in a body-fixed reference frame are, [18]:

$$\dot{\boldsymbol{\xi}} = \boldsymbol{T}_\Theta(\boldsymbol{\xi})\boldsymbol{\zeta} \tag{1a}$$

$$\boldsymbol{M}(\boldsymbol{\theta})\dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{\theta},\boldsymbol{\zeta})\boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta})\boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{\theta},\boldsymbol{\eta}) = \boldsymbol{\tau} + \boldsymbol{\tau}^{\text{ext}} \tag{1b}$$

where $\boldsymbol{T}_\Theta(\boldsymbol{\xi})$ is the transformation from body-fixed velocities to the inertial frame, $\boldsymbol{M}(\boldsymbol{\theta})$ is the inertia matrix, consisting of rigid body inertia and added mass inertia from hydrodynamic effects, $\boldsymbol{C}(\boldsymbol{\theta},\boldsymbol{\zeta})$ is the centripetal and Coriolis matrix, $\boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta})$ is the hydrodynamic damping matrix, $\boldsymbol{g}(\boldsymbol{\theta},\boldsymbol{\eta})$ contains buoyancy and gravitational forces. $\boldsymbol{\tau}$ is the vector of applied forces, while $\boldsymbol{\tau}^{\text{ext}}$ is the vector of external forces affecting the system. For systems on land $\boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta})$ is the friction matrix. VMSs are generally described by a base, and one or several manipulators with a total of n joints, where the base and all manipulator links are rigid bodies. The pose and velocities for the system base are given by:

$$\boldsymbol{\eta}^T = \begin{bmatrix}\boldsymbol{p}^T & \boldsymbol{\Theta}^T\end{bmatrix} \in \mathbb{R}^6, \quad \boldsymbol{\nu}^T = \begin{bmatrix}\boldsymbol{v}^T & \boldsymbol{\omega}^T\end{bmatrix} \in \mathbb{R}^6 \tag{2}$$

with $\boldsymbol{p} \in \mathbb{R}^3$ being the position given in an inertial frame, $\boldsymbol{\Theta} \in \mathbb{R}^3$ being the rotation parametrized with Euler angles, and $\boldsymbol{v}, \boldsymbol{\omega} \in \mathbb{R}^3$ are the body-fixed linear and angular velocities respectively. The full configuration of a VMS with n links and $n-1$ joints will have $m = 6+n-1$ DoFs, and is given by combining (2) and the joint angles $\boldsymbol{\theta} \in \mathbb{R}^{n-1}$:

$$\boldsymbol{\xi}^T = \begin{bmatrix}\boldsymbol{\eta}^T & \boldsymbol{\theta}^T\end{bmatrix} \in \mathbb{R}^m \quad \boldsymbol{\zeta}^T = \begin{bmatrix}\boldsymbol{\nu}^T & \dot{\boldsymbol{\theta}}^T\end{bmatrix} \in \mathbb{R}^m \tag{3}$$

As long as all bodies in the UVMS are symmetric, the model can be parametrized such that it has the following beneficial properties [19]

1) $\boldsymbol{M}(\boldsymbol{\theta}) = \boldsymbol{M}(\boldsymbol{\theta})^T > 0$
2) $\dot{\boldsymbol{M}}(\boldsymbol{\theta}) - 2\boldsymbol{C}(\boldsymbol{\theta},\boldsymbol{\zeta})$ is skew symmetric
3) $\boldsymbol{z}^T\boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta})\boldsymbol{z} > \boldsymbol{0} \quad \forall \boldsymbol{z} \in \mathbb{R}^m$
4) $\|\boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta}_a) - \boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta}_b)\| \leq D_M\|\boldsymbol{\zeta}_a - \boldsymbol{\zeta}_b\|$

By [18], $\boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta})$ can be split into a linear part $\boldsymbol{D}_L(\boldsymbol{\theta},\boldsymbol{\zeta})$ and a nonlinear part $\boldsymbol{d}_N(\boldsymbol{\theta},\boldsymbol{\zeta})$. For the control design we will make the following assumption.

Assumption 1: The nonlinear damping $\boldsymbol{d}_N(\boldsymbol{\theta}, \boldsymbol{\zeta}) \approx \mathbf{0}$

Remark 1: This is a realistic assumption since the nonlinear damping will be negligible during intervention operations. Moreover, we do not want to introduce a cancellation of nonlinear damping terms in our control law, since these model parameters are generally poorly known and the damping is dissipative. Cancelling these terms thus comes with a risk of introducing destabilizing terms.

III. Task coordinates

This section introduces the decoupled task coordinates used in the control law. As in [13], the control law relies on the following assumptions:

Assumption 2: The total task dimension is equal to the number of degrees of freedom m of the system.

Assumption 3: The tasks are all simultaneously feasible.

Assumption 4: The considered trajectories avoid any potential singularities.

These requirements must be met when the tasks are specified, either by a human designer or a higher level planning algorithm.

Remark 2: Note that tasks as used in this paper means a quantity given by a function of the generalized coordinates, that can follow a reference. More complex operations such as opening a door or manipulating an object requires defining suitable tasks and generating references for them. To pick up an object, one could for example have end effector position as the main task, position of a camera on the system base as a secondary task, and energy efficient joint configuration as a third. Higher level planning such as this is not considered in this paper.

For the completion of r prioritized tasks, with dimensions m_i, the task space positions are defined as functions of $\boldsymbol{\xi}$

$$\boldsymbol{x}_i = \boldsymbol{f}_i(\boldsymbol{\xi}) \tag{4}$$

for $i = 1 \dots r$. This gives the *task Jacobians* defined as

$$\boldsymbol{J}_i'(\boldsymbol{\xi}) = \frac{\partial \boldsymbol{f}_i(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \tag{5}$$

To account for the transformation between body-fixed and inertial frames, we define

$$\boldsymbol{J}_i(\boldsymbol{\xi}) = \boldsymbol{J}_i'(\boldsymbol{\xi}) \boldsymbol{T}_\Theta(\boldsymbol{\xi}) \tag{6}$$

which can be stacked to obtain the *augmented Jacobian matrix* on task level i

$$\boldsymbol{J}_i^{\text{aug}}(\boldsymbol{\xi}) = \begin{bmatrix} \boldsymbol{J}_1(\boldsymbol{\xi}) \\ \vdots \\ \boldsymbol{J}_i(\boldsymbol{\xi}) \end{bmatrix} \tag{7}$$

giving the task space velocities

$$\dot{\boldsymbol{x}}_i = \boldsymbol{J}_i'^{\text{aug}}(\boldsymbol{\xi}) \dot{\boldsymbol{\xi}} = \boldsymbol{J}_i^{\text{aug}}(\boldsymbol{\xi}) \boldsymbol{\zeta} \tag{8}$$

and augmented task space velocities and accelerations

$$\dot{\boldsymbol{x}}_i^{\text{aug}} = \begin{bmatrix} \dot{\boldsymbol{x}}_1 \\ \vdots \\ \dot{\boldsymbol{x}}_i \end{bmatrix}, \quad \ddot{\boldsymbol{x}}_i^{\text{aug}} = \begin{bmatrix} \ddot{\boldsymbol{x}}_1 \\ \vdots \\ \ddot{\boldsymbol{x}}_i \end{bmatrix} \tag{9}$$

The task space position errors are then defined as

$$\tilde{\boldsymbol{x}}_i(t) = \boldsymbol{x}_i(t) - \boldsymbol{x}_{i,d}(t) \tag{10}$$

where $\boldsymbol{x}_{i,d}(t)$ is the desired task space position at time t.

To avoid high priority tasks being affected by lower priorities, lower level tasks are solved in the null space of all higher levels, utilizing the remaining DoFs of the system. To this end, the dynamically consistent null space projector from [4] is used.

$$\boldsymbol{N}_i(\boldsymbol{\xi}) = \begin{cases} \boldsymbol{I} & \text{for } i = 1 \\ \boldsymbol{I} - \boldsymbol{J}_{i-1}^{\text{aug}}(\boldsymbol{\xi})^T \boldsymbol{J}_{i-1}^{\text{aug}}(\boldsymbol{\xi})^{M+,T} & \text{for } 1 < i \leq r \end{cases} \tag{11}$$

where $^{M+}$ denotes the dynamically consistent pseudoinverse [1]. The null space projectors are used to define the extended task Jacobians as

$$\bar{\boldsymbol{J}}_i(\boldsymbol{\xi}) = \boldsymbol{J}_i(\boldsymbol{\xi}) \boldsymbol{N}_i(\boldsymbol{\xi})^T \tag{12}$$

which can be used to define decoupled task space velocities

$$\boldsymbol{v}_i = \bar{\boldsymbol{J}}_i(\boldsymbol{\xi}) \boldsymbol{\zeta} = \underbrace{\bar{\boldsymbol{J}}_i(\boldsymbol{\xi}) \boldsymbol{J}_i^{\text{aug}}(\boldsymbol{\xi})}_{\boldsymbol{B}_i(\boldsymbol{\xi})} \dot{\boldsymbol{x}}_i \tag{13}$$

that are stacked to get the complete task space velocity vector $\boldsymbol{v} \in \mathbb{R}^n$, and the corresponding Jacobian matrix $\bar{\boldsymbol{J}}(\boldsymbol{\xi}) \in \mathbb{R}^{n \times n}$. Inserting

$$\boldsymbol{v} = \bar{\boldsymbol{J}}(\boldsymbol{\xi}) \boldsymbol{\zeta} \quad \text{and} \quad \dot{\boldsymbol{v}} = \dot{\bar{\boldsymbol{J}}}(\boldsymbol{\xi}) \boldsymbol{\zeta} + \bar{\boldsymbol{J}}(\boldsymbol{\xi}) \dot{\boldsymbol{\zeta}} \tag{14}$$

into (1) yields the decoupled equations of motion (dependencies omitted for clarity)

$$\begin{aligned} \bar{\boldsymbol{J}}^{-T} \boldsymbol{M} \bar{\boldsymbol{J}}^{-1} \dot{\boldsymbol{v}} = \bar{\boldsymbol{J}}^{-T} (\boldsymbol{\tau} + \boldsymbol{\tau}^{\text{ext}} - \boldsymbol{g}) \\ + \bar{\boldsymbol{J}}^{-T} (-\boldsymbol{C} - \boldsymbol{D}) \bar{\boldsymbol{J}}^{-1} \boldsymbol{v} \\ + \bar{\boldsymbol{J}}^{-T} \boldsymbol{M} \bar{\boldsymbol{J}}^{-1} \dot{\bar{\boldsymbol{J}}} \bar{\boldsymbol{J}}^{-1} \boldsymbol{v} \end{aligned} \tag{15}$$

We define the transformed inertia matrix as

$$\bar{\boldsymbol{M}} = \bar{\boldsymbol{J}}^{-T} \boldsymbol{M} \bar{\boldsymbol{J}}^{-1} \tag{16}$$

giving

$$\begin{aligned} \bar{\boldsymbol{M}} \dot{\boldsymbol{v}} = \bar{\boldsymbol{J}}^{-T} (\boldsymbol{\tau} + \boldsymbol{\tau}^{\text{ext}} - \boldsymbol{g}) \\ - \underbrace{\bar{\boldsymbol{J}}^{-T} \left(\boldsymbol{C} - \boldsymbol{M} \bar{\boldsymbol{J}}^{-1} \dot{\bar{\boldsymbol{J}}} \right) \bar{\boldsymbol{J}}^{-1}}_{\boldsymbol{\mu}} \boldsymbol{v} \\ - \underbrace{\bar{\boldsymbol{J}}^{-T} (\boldsymbol{D}_L) \bar{\boldsymbol{J}}^{-1}}_{\boldsymbol{\delta}} \boldsymbol{v} \end{aligned} \tag{17}$$

such that the matrix $\boldsymbol{\mu}$ contains the transformed Coriolis terms and $\boldsymbol{\delta}$ the transformed linear damping. The system in decoupled coordinates then takes the form:

$$\bar{\boldsymbol{M}} \dot{\boldsymbol{v}} = \bar{\boldsymbol{J}}^{-T} (\boldsymbol{\tau} + \boldsymbol{\tau}^{\text{ext}} - \boldsymbol{g}) - (\boldsymbol{\mu} + \boldsymbol{\delta}) \boldsymbol{v} \tag{18}$$

The transformed inertia matrix $\bar{\boldsymbol{M}}$ is block diagonal, $\bar{\boldsymbol{M}} = \text{diag}\left(\bar{\boldsymbol{M}}_1, ..., \bar{\boldsymbol{M}}_r\right)$, where $\bar{\boldsymbol{M}}_i \in R^{m_i \times m_i}$, meaning that the tasks are decoupled at the inertial level. However, $\boldsymbol{\mu}$ and $\boldsymbol{\delta}$ still have off-diagonal terms, meaning that there is some coupling left between tasks that must be dealt with by the control law.

IV. CONTROL LAW

The control law is chosen as

$$\boldsymbol{\tau} = \boldsymbol{g} + \boldsymbol{\tau}_\mu + \boldsymbol{\tau}_\delta + \sum_{i=1}^{r} \boldsymbol{N}_i \boldsymbol{J}_i^T \boldsymbol{F}_{i,\text{ctrl}} \tag{19}$$

where $\boldsymbol{g}$ compensates for the buoyancy and gravity forces, $\boldsymbol{\tau}_\mu$ cancels off-diagonal terms in $\boldsymbol{\mu}$, and is defined as

$$\boldsymbol{\tau}_\mu = \sum_{i=1}^{r} \left(\bar{\boldsymbol{J}}_i^T \left(\sum_{j=1}^{i-1} \boldsymbol{\mu}_{i,j} \boldsymbol{v}_j + \sum_{j=i+1}^{r} \boldsymbol{\mu}_{i,j} \boldsymbol{v}_j \right) \right) \tag{20}$$

and $\boldsymbol{\tau}_\delta$ is defined equivalently to $\boldsymbol{\tau}_\mu$, but cancelling the off-diagonal damping terms.

Remark 3: For systems without hydrodynamic effects, this simplifies to the expression given in [13]:

$$\boldsymbol{\tau} = \boldsymbol{g} + \boldsymbol{\tau}_\mu + \sum_{i=1}^{r} \boldsymbol{N}_i \boldsymbol{J}_i^T \boldsymbol{F}_{i,\text{ctrl}} \tag{21}$$

Applying the control law (19) to (18) gives, for $i = 1, \ldots, r$

$$\bar{\boldsymbol{M}}_i \dot{\boldsymbol{v}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i}) \boldsymbol{v}_i = \boldsymbol{F}_{i,\text{ctrl}} + \boldsymbol{F}_{v_i}^{\text{ext}} \tag{22}$$

and in task space coordinates:

$$\bar{\boldsymbol{M}} \ddot{\boldsymbol{x}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i}) \dot{\boldsymbol{x}}_i + \boldsymbol{\gamma}_i \begin{bmatrix} \dot{\boldsymbol{x}}_{i-1}^{\text{aug}} \\ \ddot{\boldsymbol{x}}_{i-1}^{\text{aug}} \end{bmatrix} = \boldsymbol{F}_{i,\text{ctrl}} + \boldsymbol{F}_{v_i}^{\text{ext}} \tag{23}$$

where

$$\boldsymbol{\gamma}_i(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) = [\boldsymbol{\Gamma}_{\boldsymbol{i,1}}, \ldots, \boldsymbol{\Gamma}_{\boldsymbol{i,i-1}} \boldsymbol{\Psi}_{i,1}, \ldots, \boldsymbol{\Psi}_{i,i-1}] \tag{24a}$$

$$\boldsymbol{\Gamma}_{i,j} = \boldsymbol{u}_{i,i} \boldsymbol{B}_{i,j} + \bar{\boldsymbol{M}}_i \dot{\boldsymbol{B}}_{i,j}, \quad \boldsymbol{\Psi}_{i,j} = \bar{\boldsymbol{M}}_i \boldsymbol{B}_{i,j} \tag{24b}$$

contains top-down disturbances caused by higher prioritized tasks, $\boldsymbol{\mu}_{i,i}$ is the i-th element on the diagonal of the Coriolis matrix, $\boldsymbol{\delta}_{i,i}$ the i-th element on the diagonal of the transformed damping matrix. Note that the transformed matrices still retain the skew symmetry property from (1), i.e. $\frac{1}{2}\dot{\bar{\boldsymbol{M}}}_i - \boldsymbol{\mu}_{i,i}$ is skew symmetric [13]. $\boldsymbol{F}_{i,\text{ctrl}}$ is the control force on level i, and $\boldsymbol{F}_{v_i}^{\text{ext}}$ is the generalized external force on the i-th task in the decoupled coordinates

$$\boldsymbol{F}_{v_i}^{\text{ext}} = \bar{\boldsymbol{J}}_i^{-T} \boldsymbol{\tau}^{\text{ext}} \tag{25}$$

In the following, we will explore three alternatives for the control force $\boldsymbol{F}_{i,\text{ctrl}}$; one equal to the one used in [13] except for the addition of a hydrodynamic damping term, one utilizing a sliding variable, and one using sliding mode control (SMC). We will show that the sliding variable and SMC approaches give better stability and robustness properties in the presence of disturbances and modelling errors, a requirement for underwater systems.

The first alternative control law is given by

$$\begin{aligned} \boldsymbol{F}_{i,\text{ctrl}} = \bar{\boldsymbol{M}}_i \ddot{\boldsymbol{x}}_{i,\text{des}} + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i}) \dot{\boldsymbol{x}}_{i,\text{des}} \\ - \boldsymbol{D}_{\boldsymbol{i}} \dot{\tilde{\boldsymbol{x}}}_i - \boldsymbol{K}_{\boldsymbol{i}} \tilde{\boldsymbol{x}}_i + \boldsymbol{\gamma}_i \begin{bmatrix} \dot{\boldsymbol{x}}_{i-1,\text{des}}^{\text{aug}} \\ \ddot{\boldsymbol{x}}_{i-1,\text{des}}^{\text{aug}} \end{bmatrix} \end{aligned} \tag{26}$$

where $\boldsymbol{D}_{\boldsymbol{i}}$ and $\boldsymbol{K}_{\boldsymbol{i}}$ are the desired damping and stiffness matrices, respectively. They are chosen such that $\boldsymbol{D}_i = \boldsymbol{D}_i^T > \boldsymbol{0}$ and $\boldsymbol{K}_i = \boldsymbol{K}_i^T > \boldsymbol{0}$. This gives the closed-loop system on task level i:

$$\begin{aligned} \bar{\boldsymbol{M}}_i \ddot{\tilde{\boldsymbol{x}}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{\boldsymbol{i},\boldsymbol{i}} + \boldsymbol{D}_i) \dot{\tilde{\boldsymbol{x}}}_i + \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i \\ = -\boldsymbol{\gamma}_i \begin{bmatrix} \dot{\tilde{\boldsymbol{x}}}_{i-1}^{\text{aug}} \\ \ddot{\tilde{\boldsymbol{x}}}_{i-1}^{\text{aug}} \end{bmatrix} + \boldsymbol{F}_{v_i}^{\text{ext}} \end{aligned} \tag{27}$$

The second alternative control law, motivated by [20], [21], and [22], utilizes a sliding variable:

$$\begin{aligned} \boldsymbol{F}_{i,\text{ctrl}} = \bar{\boldsymbol{M}}_i \ddot{\boldsymbol{x}}_{i,r} + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i}) \dot{\boldsymbol{x}}_{i,r} \\ - \boldsymbol{D}_i \boldsymbol{s}_i - \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i + \boldsymbol{\gamma}_i \begin{bmatrix} \dot{\boldsymbol{x}}_{i-1,\text{des}}^{\text{aug}} \\ \ddot{\boldsymbol{x}}_{i-1,\text{des}}^{\text{aug}} \end{bmatrix} \end{aligned} \tag{28}$$

with

$$\boldsymbol{s}_i = \dot{\tilde{\boldsymbol{x}}}_i + \boldsymbol{\Lambda}_i \tilde{\boldsymbol{x}}_i, \qquad \boldsymbol{\Lambda}_i = \boldsymbol{\Lambda}_i^T > \boldsymbol{0} \tag{29}$$

and also differs from (26) by $\dot{\boldsymbol{x}}_{i,\text{des}}$ being replaced by the reference velocity

$$\dot{\boldsymbol{x}}_{i,r} = \dot{\boldsymbol{x}}_{i,\text{des}} - \boldsymbol{\Lambda}_i \tilde{\boldsymbol{x}}_i \tag{30}$$

The sliding variable $\boldsymbol{s}$ introduced in [20] can be interpreted as the velocity error with respect to the new reference velocity:

$$\boldsymbol{s}_i = \dot{\boldsymbol{x}}_i - \dot{\boldsymbol{x}}_{i,r} \tag{31}$$

The resulting closed-loop system is then

$$\begin{aligned} \bar{\boldsymbol{M}}_i \dot{\boldsymbol{s}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i} + \boldsymbol{D}_i) \boldsymbol{s}_i + \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i \\ = -\boldsymbol{\gamma}_i \begin{bmatrix} \dot{\tilde{\boldsymbol{x}}}_{i-1}^{\text{aug}} \\ \ddot{\tilde{\boldsymbol{x}}}_{i-1}^{\text{aug}} \end{bmatrix} + \boldsymbol{F}_{v_i}^{\text{ext}} \end{aligned} \tag{32a}$$

$$\dot{\tilde{\boldsymbol{x}}}_i = -\boldsymbol{\Lambda}_i \tilde{\boldsymbol{x}}_i + \boldsymbol{s}_i \tag{32b}$$

One motivation for the choice of control law (28) is that, as we will show, it improves the convergence properties and significantly simplifies the stability analysis compared to [16]. Moreover, as the control law and corresponding analysis are expressed using the sliding mode variable s, they can readily be extended to further increase the robustness through sliding mode control in **the third alternative control law**:

$$\begin{aligned} \boldsymbol{F}_{i,\text{ctrl}} = \bar{\boldsymbol{M}}_i \ddot{\boldsymbol{x}}_{i,r} + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i}) \dot{\boldsymbol{x}}_{i,r} - \boldsymbol{D}_i \boldsymbol{s}_i - \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i \\ - \boldsymbol{K}_{S,i} \text{sgn}(\boldsymbol{s}_i) + \boldsymbol{\gamma}_i \begin{bmatrix} \dot{\boldsymbol{x}}_{i-1,\text{des}}^{\text{aug}} \\ \ddot{\boldsymbol{x}}_{i-1,\text{des}}^{\text{aug}} \end{bmatrix} \end{aligned} \tag{33}$$

where $\boldsymbol{K}_{S,i} = \boldsymbol{K}_{S,i}^T > \boldsymbol{0}$ is the sliding mode gain and $\text{sgn}(\boldsymbol{s}_i)$ is a vector containing the signs of $\boldsymbol{s}_i$. This gives the closed-loop system

$$\begin{aligned} \bar{\boldsymbol{M}}_i \dot{\boldsymbol{s}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i} + \boldsymbol{D}_i) \boldsymbol{s}_i + \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i + \boldsymbol{K}_{s,i} \text{sgn}(\boldsymbol{s}_i) \\ = -\boldsymbol{\gamma}_i \begin{bmatrix} \dot{\tilde{\boldsymbol{x}}}_{i-1}^{\text{aug}} \\ \ddot{\tilde{\boldsymbol{x}}}_{i-1}^{\text{aug}} \end{bmatrix} + \boldsymbol{F}_{v_i}^{\text{ext}} \end{aligned} \tag{34a}$$

$$\dot{\tilde{\boldsymbol{x}}}_i = -\boldsymbol{\Lambda}_i \tilde{\boldsymbol{x}}_i + \boldsymbol{s}_i \tag{34b}$$

Remark 4 (Cancellation of external forces): If measurements of external forces are available, then the control law can be extended to include external force feedback. This is done in [13] and used in [23] to fully compensate for the top-down disturbances. However, as this will typically not be possible for underwater systems, and specifically is not available for the considered AIAUV, it will not be discussed further here.

V. STABILITY ANALYSIS

We consider the case of free motion ($\boldsymbol{F}^{\text{ext}} = \boldsymbol{\tau}^{\text{ext}} = \mathbf{0}$), and define $\boldsymbol{y}$ and the sets $\mathcal{A}_i$ analogously to [13, Section IV.A]. The state vector $\boldsymbol{y} \in \mathbb{R}^{2n}$ is defined as

$$\boldsymbol{y}_i = \begin{bmatrix} \tilde{\boldsymbol{x}}_i \\ \boldsymbol{s}_i \end{bmatrix}, \boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y_r} \end{bmatrix} \tag{35}$$

representing task-space position, and velocity errors of all tasks. The stability of the equilibrium point $\boldsymbol{y} = \mathbf{0}$, where all task-space errors are $\mathbf{0}$, will be examined. The *nested sets for the successive accomplishment of tasks* [13] are defined with $\mathcal{A}_0$ being the full state space of the system, and

$$\mathcal{A}_1 = \{\boldsymbol{y} \mid \tilde{\boldsymbol{x}}_1 = \mathbf{0}, \boldsymbol{s}_1 = \mathbf{0}\} \tag{36}$$

being the subset of the task space where the highest prioritized task is fulfilled. Sets $\mathcal{A}_2 ... \mathcal{A}_r$ are defined recursively as subsets of the previous set with one more completed task.

$$\mathcal{A}_i = \mathcal{A}_{i-1} \bigcap \{\boldsymbol{y} \mid \tilde{\boldsymbol{x}}_i = \mathbf{0}, \boldsymbol{s}_i = \mathbf{0}\} \tag{37}$$

A. Stability analysis with control law (26)

For $\boldsymbol{y} \in \mathcal{A}_{i-1}$ in free motion ($\boldsymbol{F}^{\text{ext}} = \boldsymbol{\tau}^{\text{ext}} = \mathbf{0}$), the closed-loop task error dynamics on priority level i becomes

$$\bar{\boldsymbol{M}}_i \ddot{\tilde{\boldsymbol{x}}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta_{i,i}} + \boldsymbol{D}_i)\dot{\tilde{\boldsymbol{x}}}_i + \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i = \mathbf{0} \tag{38}$$

The following theorem follows directly from the analysis of [13]. Specifically, it is straightforward to include the hydrodynamic damping terms in the analysis of [13], and defining $\boldsymbol{y}^*$ as

$$\boldsymbol{y}_i^* = \begin{bmatrix} \tilde{\boldsymbol{x}}_i \\ \dot{\tilde{\boldsymbol{x}}}_i \end{bmatrix}, \boldsymbol{y}^* = \begin{bmatrix} \boldsymbol{y}_1^* \\ \vdots \\ \boldsymbol{y_r^*} \end{bmatrix} \tag{39}$$

to prove the following result:

Theorem 1: Consider the closed-loop task error dynamics of system (1) with control law (19), (26). For free motion ($\boldsymbol{F}^{\text{ext}} = \boldsymbol{\tau}^{\text{ext}} = \mathbf{0}$), the origin $\boldsymbol{y}^* = \mathbf{0}$ of the closed-loop task error dynamics is UAS.

B. Stability analysis with control law (28)

In this section, we consider the closed-loop task error dynamics (32). For the case of free motion, and for $\boldsymbol{y} \in \mathcal{A}_{i-1}$, the closed-loop task error dynamics on task level i becomes

$$\bar{\boldsymbol{M}}_i \dot{\boldsymbol{s}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i} + \boldsymbol{D}_i)\boldsymbol{s}_i + \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i = \mathbf{0} \tag{40a}$$

$$\dot{\tilde{\boldsymbol{x}}}_i = -\boldsymbol{\Lambda}_i \tilde{\boldsymbol{x}}_i + \boldsymbol{s}_i \tag{40b}$$

Theorem 2: Consider the closed-loop task error dynamics consisting of system (1) with control law (19), (28) with $\boldsymbol{\Lambda}_i = \lambda_i \boldsymbol{I}, \lambda_i > 0, i = 1, ..., r$. For free motion ($\boldsymbol{F}^{\text{ext}} = \boldsymbol{\tau}^{\text{ext}} = \mathbf{0}$), the origin $\boldsymbol{y} = \mathbf{0}$ of the closed-loop task error dynamics is UAS. Moreover, the closed-loop task dynamics of the highest priority task is ES and the task dynamics of each lower priority task is exponentially stable when all higher level tasks are fulfilled.

Proof: To analyze the stability of the closed-loop task error dynamics on priority level i, we use the Lyapunov function candidate

$$V_i(\boldsymbol{s}_i, \tilde{\boldsymbol{x}}_i) = \frac{1}{2}\boldsymbol{s}_i^T \bar{\boldsymbol{M}}_i \boldsymbol{s}_i + \frac{1}{2}\tilde{\boldsymbol{x}}_i^T \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i \tag{41}$$

and its time deritive, given by:

$$\dot{V}_i = \frac{1}{2}\boldsymbol{s}_i^T \dot{\bar{\boldsymbol{M}}}_i \boldsymbol{s}_i + \boldsymbol{s}_i^T \bar{\boldsymbol{M}}_i \dot{\boldsymbol{s}}_i + \tilde{\boldsymbol{x}}_i^T \boldsymbol{K_i} \dot{\tilde{\boldsymbol{x}}}_i \tag{42}$$

Inserting (40), and utilizing that $\frac{1}{2}\dot{\bar{\boldsymbol{M}}}_i - \boldsymbol{\mu}_{i,i}$ is skew-symmetric, gives:

$$\dot{V}_i = -\boldsymbol{s}_i^T (\boldsymbol{D}_i + \boldsymbol{\delta}_{i,i})\boldsymbol{s}_i - \tilde{\boldsymbol{x}}_i^T \boldsymbol{\Lambda}_i \boldsymbol{K}_i \tilde{\boldsymbol{x}}_i \tag{43}$$

We choose $\boldsymbol{\Lambda}_i = \lambda_i \boldsymbol{I}$ with $\lambda_i > 0$ constant. Using $\lambda_{\min}$ and $\lambda_{\max}$ as minimum and maximum eigenvalues of a matrix, by [24, Theorem 4.10] the origin of (40) is ES, with:

$$k_{1,i} = \frac{1}{2}\min\{\lambda_{\min}(\bar{\boldsymbol{M}}_i), \lambda_{\min}(\boldsymbol{K}_i)\} \tag{44a}$$

$$k_{2,i} = \frac{1}{2}\max\{\lambda_{\max}(\bar{\boldsymbol{M}}_i), \lambda_{\max}(\boldsymbol{K}_i)\} \tag{44b}$$

$$k_{3,i} = \min\{\lambda_{\min}(D_m(\boldsymbol{D}_i + \boldsymbol{\delta}_{i,i})), \lambda_i \cdot \lambda_{\min}(\boldsymbol{K}_i)\} \tag{44c}$$

To analyze the stability of the equilibrium point $\boldsymbol{y} = \mathbf{0}$ of the total closed-loop task error dynamics including all priority levels, we will use [25, Theorem 2], cf. Appendix A. Note that since (41) is a strict Lyapunov function of the system (40), the proof becomes quite a lot simpler than the corresponding proof in [13], avoiding the two-step process that was needed to handle a non-strict Lyapunov function.

1) We start by analyzing the error dynamics of the lowest priority task, $i = r$, within $\mathcal{A}_{r-1}$ (i.e. when all tasks are completed except the lowest prioritized). By the above analysis $\boldsymbol{y}_r = \mathbf{0}$ is ES, which implies by Definition A.1 that the origin of the total system $\boldsymbol{y} = \mathbf{0}$ is $\mathcal{A}_{r-1}$-UAS
2) For $i = r - 1$, within $\mathcal{A}_{r-2}$, by the above analysis, $\dot{V}_i$ is negative definite, and with $k_{1,i}, k_{2,i}, k_{3,i}$ defined by (44) (A.2) is satisfied with $W_1 = k_{1,i}\|y\|^2$, $W_2 = k_{2,i}\|y\|^2$ and (A.3) is satisfied with $\alpha(W_1) = \frac{k_{3,i}}{k_{1,i}} W_1$. Along with the fact from the previous step that the origin, $\boldsymbol{y} = \mathbf{0}$, is $\mathcal{A}_i$-UAS, we can use Theorem A.1 to conclude that it is also is $\mathcal{A}_{i-1}$-UAS
3) Step 2 is repeated recursively for each i until we reach $i = 1$. We now have that $\boldsymbol{y} = \mathbf{0}$ is $\mathcal{A}_1$-UAS
4) In the complete task space $\mathcal{A}_0$, with the origin being $\mathcal{A}_1$-UAS and $\dot{V}_1$ negative definite, we can use Theorem A.1 again, and conclude that the origin of the complete system, $\boldsymbol{y} = \mathbf{0}$, is UAS.
5) Moreover, choosing $k_{1,1}$, $k_{2,1}$ and $k_{3,1}$ as in (44), we can conclude exponential stability (ES) of the highest priority task dynamics by [24, Theorem 4.10]

Thus, the equilibrium point $\boldsymbol{y} = \mathbf{0}$, where all tasks are fulfilled, is UAS. The task dynamics of the highest priority task is also exponentially stable, and the dynamics of each task is exponentially stable when all higher level tasks are fulfilled. This implies that the highest prioritized task will always converge exponentially, and that each task will converge exponentially when all higher level tasks are fulfilled. ■

Remark 5: As exponential stability in addition to improved convergence properties also provides stronger robustness properties than uniform asymptotic stability, cf. [24, Lemmas 9.2 and 9.3] we can thus expect that the proposed control law (28) provides both improved convergence and robustness compared to (26).

Remark 6: Note that the inherent system damping, $\boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta})\boldsymbol{\zeta}$, in (1) is not required for the closed-loop system analysis to hold. The inherent damping only improves the convergence rate, cf. (44), and the resulting compliant behavior is characterized by $\boldsymbol{D}_{\text{total},i} = \boldsymbol{D}_i + \boldsymbol{\delta}_{i,i}$, but the analysis also holds for systems (1) where $\boldsymbol{D}(\boldsymbol{\theta},\boldsymbol{\zeta})\boldsymbol{\zeta} = \boldsymbol{0}$.

C. Stability analysis with control law (33)

Using the control force (33) in free motion without disturbances, the same Lyapunov function candidate V_i as in (41) can be used with the only difference that the time-derivative of V_i has an extra sliding mode term

$$\begin{aligned}\dot{V}_i &= -\boldsymbol{s}_i^T(\boldsymbol{D}_i + \boldsymbol{\delta}_{i,i})\boldsymbol{s}_i - \boldsymbol{s}_i^T\boldsymbol{K}_{S,i}\text{sgn}(\boldsymbol{s}_i) - \tilde{\boldsymbol{x}}_i^T\boldsymbol{\Lambda}_i\boldsymbol{K}_i\tilde{\boldsymbol{x}}_i \\ &\leq -\boldsymbol{s}_i^T(\boldsymbol{D}_i + \boldsymbol{\delta}_{i,i})\boldsymbol{s}_i - \tilde{\boldsymbol{x}}_i^T\boldsymbol{\Lambda}_i\boldsymbol{K}_i\tilde{\boldsymbol{x}}_i \end{aligned} \tag{45}$$

Thus, the analysis from Section V-B still holds, and Theorem 2 holds with (28) replaced by (33) if a Lipschitz continuous approximation of the sign term is used (as in [24, Chaper 14.2]). However, we want to show robustness. Therefore, we consider the system perturbed by modeling errors and other state- and time-varying disturbances, for which the resulting closed-loop level i task error dynamics for free motion and for $\boldsymbol{y} \in \mathcal{A}_{i-1}$ can be described by

$$\begin{aligned}\bar{\boldsymbol{M}}_i\dot{\boldsymbol{s}}_i + (\boldsymbol{\mu}_{i,i} + \boldsymbol{\delta}_{i,i} + \boldsymbol{D}_i)\boldsymbol{s}_i + \boldsymbol{K}_i\tilde{\boldsymbol{x}}_i \\ + \boldsymbol{K}_{S,i}\text{sgn}(\boldsymbol{s}_i) + \boldsymbol{f}_i(\boldsymbol{s}_i,\tilde{\boldsymbol{x}}_i,t) = \boldsymbol{0}\end{aligned} \tag{46a}$$

$$\dot{\tilde{\boldsymbol{x}}}_i = -\boldsymbol{\Lambda}_i\tilde{\boldsymbol{x}}_i + \boldsymbol{s}_i \tag{46b}$$

where $\boldsymbol{f}_i(\boldsymbol{s}_i,\tilde{\boldsymbol{x}}_i,t)$ is a disturbance term representing modeling errors and other state- and time-varying disturbances that satisfy

Assumption 5:

$$|\boldsymbol{f}_{i,j}(\boldsymbol{s}_i,\tilde{\boldsymbol{x}}_i,t)| \leq \boldsymbol{L}_{i,j} \quad j \in \{1,\ldots,m_i\}$$

where $\boldsymbol{f}_{i,j}$ is the j-th element of $\boldsymbol{f}_i$, and $\boldsymbol{L}_i$ is a vector of positive elements with the same dimension as $\boldsymbol{s}_i$ and $\boldsymbol{f}_i$.

Theorem 3: Consider the closed-loop task error dynamics consisting of system (1) subject to a disturbance term satisfying Assumption 5, with control law (19), (33) with $\boldsymbol{\Lambda}_i = \lambda_i\boldsymbol{I}, \lambda_i > 0, i = 1,...,r$, and $\boldsymbol{K}_{S,i}$ diagonal with $\boldsymbol{K}_{S,i,j} \geq \boldsymbol{L}_{i,j}$ For free motion ($\boldsymbol{F}^{\text{ext}} = \boldsymbol{\tau}^{\text{ext}} = \boldsymbol{0}$), the sliding variable $\boldsymbol{s}_i$ converges to zero in finite time, after which the origin of task position error $\tilde{\boldsymbol{x}}_i$ is exponentially stable.

Proof: Using the same Lyapunov function candidate V_i as in (41), the time derivative becomes

$$\begin{aligned}\dot{V}_i = -\boldsymbol{s}_i^T(\boldsymbol{D}_i + \boldsymbol{\delta}_{i,i})\boldsymbol{s}_i - \tilde{\boldsymbol{x}}_i^T\boldsymbol{\Lambda}_i\boldsymbol{K}_i\tilde{\boldsymbol{x}}_i \\ - \boldsymbol{s}_i^T\boldsymbol{K}_{S,i}\text{sgn}(\boldsymbol{s}_i) + \boldsymbol{s}_i^T\boldsymbol{f}_i(\boldsymbol{s}_i,\tilde{\boldsymbol{x}}_i,t)\end{aligned} \tag{47}$$

$$\begin{aligned}\leq -\boldsymbol{s}_i^T(\boldsymbol{D}_i + \boldsymbol{\delta}_{i,i})\boldsymbol{s}_i - \tilde{\boldsymbol{x}}_i^T\boldsymbol{\Lambda}_i\boldsymbol{K}_i\tilde{\boldsymbol{x}}_i \\ - |\boldsymbol{s}_i^T|(\text{diag}(\boldsymbol{K}_{S,i}) - \boldsymbol{L}_i)\end{aligned} \tag{48}$$

Fig. 1. The Eelume 500 AIAUV in a "C-shape" (Courtesy of Eelume)

Where $|\boldsymbol{s}_i^T|$ is the elementwise absolute values of $\boldsymbol{s}_i$, and $\text{diag}(\boldsymbol{K}_{S,i})$ is a vector consisting of the elements on the diagonal of $\boldsymbol{K}_{S,i}$. By choosing $\boldsymbol{K}_{S,i,j} > \boldsymbol{L}_{i,j}$, $\boldsymbol{s}_i$ thus converges to zero in finite time by the comparison principle [24]. The dynamics of the position error are given in (46b). This is a linear system that is clearly ISS in regard to $\boldsymbol{s}_i$ and ES when $\boldsymbol{s}_i = \boldsymbol{0}$. ■

Remark 7: Theorem A.1 cannot be used in this case as it requires a Lipschitz system. An equivalent theorem for non-Lipschitz systems is not known to us, and therefore stability for the total system can not be rigorously proven even with improved robustness for the dynamics of each task. However, the simulations in Section VI shows that the control law behaves well in practice.

Remark 8: Note that the assumption on the disturbance term is an assumption on the modelling errors and disturbances only. This is in contrast to similar assumptions for sliding mode control approaches where no knowledge of the structure and properties of the model are utilized, and which instead aim to override all the dynamics through a (higher gain) sliding mode term. This would require an assumption that the whole system dynamics are bounded to prove stability, while we here utilize the model structure and properties and consequently arrive at a milder assumption.

Remark 9: It is also notable that Assumption 5 only requires the modeling errors and disturbances that depend on same and *lower*-priority task variables to be bounded. Specifically, $\boldsymbol{f}_i$ depends only on same and lower-priority tasks (as higher-priority tasks are assumed to be satisfied.)

VI. SIMULATIONS

The proposed control laws were implemented on an Eelume 150 AIAUV in simulation. The model in the simulator consists of an elongated body with 4 z-revolute and 4 y-revolute joints, giving a total of $m = 14$ DoFs. It can be thought of as a manipulator arm with no separate base, however, it is convenient to treat the first link as a base and use the notation for general UVMSs. The system is actuated by the joints and 7 thrusters. The thrusters can produce up to 60 N of force each, and the joints motors up to 16 Nm of torque. Further details, including model parameters can be found in [26], [9].

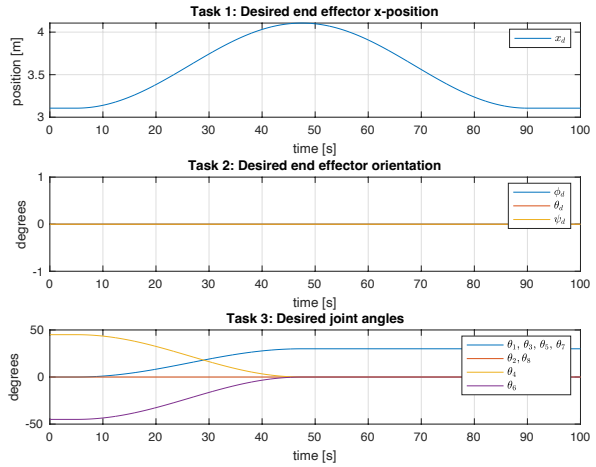

Fig. 2. Desired task trajectories. (Note that $y_d = z_d = 0$ at all times)

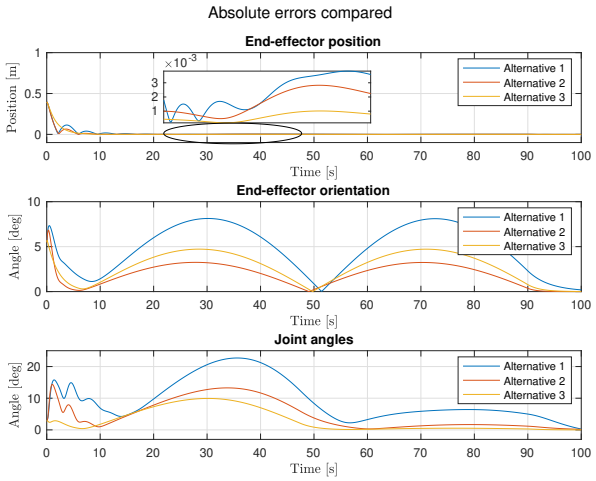

Fig. 3. Absolute errors of all tasks compared for the alternative control laws

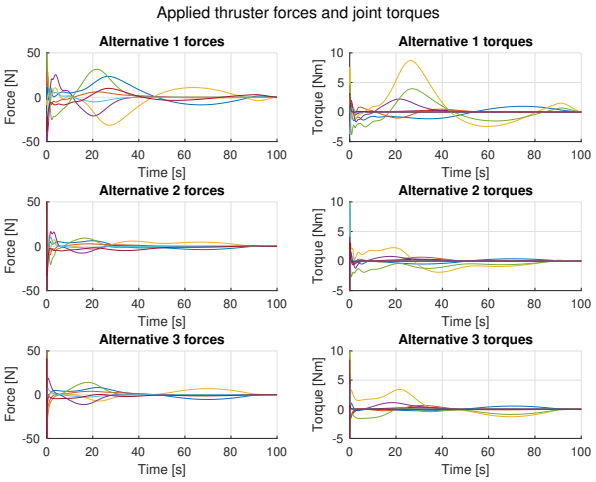

Fig. 4. Applied thruster forces and joint torques for the control laws

In the simulation, a task hierarchy with three different tasks were chosen; end-effector position, end-effector orientation and joint angles. The initial reference configuration of the system at rest has its base in the world frame origin, and 45 deg angles in the y-revolute joints. The desired trajectories are shown in Fig. 2, and are defined such that the end-effector should move smoothly 1 meter forwards and back again while keeping its initial orientation. Simultaneously, the joints should move to a "C-shape", as in Fig. 1, with 30 deg angles in the z-revolute joints.

To investigate the robustness to modelling errors, the control laws were implemented with a constant 10% error in the inertia matrix, which also impacts the Coriolis terms. The initial configuration was also chosen with 25 cm offset in x, y and z in base position and 1 deg error in all joint angles, giving a fairly substantial initial error in the end-effector position and orientation. The chosen controller parameters can be seen in Tables I and II, with the same values being used for all three alternatives. To avoid chattering, the sign term in (33) was approximated as: $\text{sgn}(s) \approx \frac{s}{|s|+0.01}$.

TABLE I

THE CONTROLLER PARAMETERS USED IN THE SIMULATION.

Task	Parameter	Value
Level 1	K_i	diag(40, 30, 30)
	D_i	diag(20, 30, 30)
Level 2	K_i	diag(10, 10, 15)
	D_i	diag(40, 40, 60)
Level 3	K_i	diag(10, 10, $\cdots$, 10)
	D_i	diag(5, 5, $\cdots$, 5)

The absolute errors of the three tasks are shown in Fig. 3, and the applied thruster forces are shown in Fig. 4. All three alternatives achieve functionally perfect tracking of the main task, with errors in the order of millimeters after the initial transient. Alternatives 1 and 2 display some oscillations before settling when compensating for the initial error.

The differences in performance on the second task are much clearer, with larger errors. This is expected as the top-down disturbances from the higher priority task affects performance. Alternative 1 does not manage to eliminate the initial error before movement starts, and has fairly large errors throughout. Notably, alternative 2 seems to perform better than alternative 3. Finally, in the joint angles task, there is a large top down disturbance that causes large initial errors for alternatives 1 and 2. Alternative 3 achieves much better performance during the transient from the initial error, and slightly better performance throughout the simulation.

In Fig. 4, the applied thruster forces and joint torques for the three alternatives are shown. All three alternatives show an initial spike in applied force, which would be less prevalent in the real system as thruster dynamics are neglected in the simulation. After the initial spike, alternative 1 applies notably higher forces and torques than alternatives 2 and 3,

TABLE II

THE SLIDING VARIABLE PARAMETERS USED IN THE SIMULATION.

Task	Parameter	Value
Level 1,2,3	Λ_i	diag(0.4, 0.4, $\cdots$, 0.4)
	$K_{S,i}$	diag(20, 20, $\cdots$, 20)

attempting to compensate for the larger errors in the two lowest prioritized tasks. However, all three alternatives stay within the limits of the actuators. Overall, alternatives 2 and 3 perform noticeably better than alternative 1, having both faster convergence and better tracking on lower priority tasks while applying lower forces and torques.

VII. CONCLUSIONS

To enable autonomous completion of subsea intervention tasks, a controller that allows strict task priorities and compliant contact behavior while being robust to disturbances and modelling errors is needed. In this paper a robust control law for vehicle-manipulator systems was developed, which allows utilizing kinematic redundancy for completion of simultaneous tasks in strict priority, with a specified compliant behavior during interaction. While developed specifically for floating-base underwater systems, the control law can also be applied to fixed-base or mobile systems on land. The resulting closed-loop system was proven to be UAS, with exponentially stable dynamics for the highest prioritized task, and to have beneficial robustness properties. The theory and the effectiveness of the proposed control law were verified in a simulation study. In the future, simulations with interaction tasks should be performed, and physical experiments conducted to further validate the control law.

APPENDIX A

Definition A.1: A system is said to be Ω-US if it is US conditional on the set Ω, that is,

$$\forall z_0 \in \Omega, \|z_0\| < \delta(\epsilon) \implies \|z_0\| < \epsilon, \quad \forall t > 0 \tag{A.1}$$

It is Ω-UAS if it is Ω-US and uniformly attractive on Ω.

We here restate [25, Theorem 2] for completeness.

Theorem A.1: For a time-varying system $\dot{\boldsymbol{x}} = \boldsymbol{f}(t, \boldsymbol{x})$ with $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{f} : \mathbb{R}_{\geq 0} \times \mathcal{D} \to \mathbb{R}^n$, where $\mathcal{D} \subset \mathbb{R}^n$ is a domain with $\boldsymbol{x} = \boldsymbol{0}$ and $\boldsymbol{f}(t, \boldsymbol{0}) \equiv \boldsymbol{0} \quad \forall t > t_0$. It is assumed that $\boldsymbol{f}$ is continuous and locally Lipschitz in $\boldsymbol{x}$ uniformly in t. If there exists a function $V(t, \boldsymbol{x}) \in \mathcal{C}^1(\mathbb{R}_{\geq t_0} \times \mathcal{D}, \mathbb{R}_{\geq 0})$, and positive semidefinite functions W_1 and W_2 such that

$$W_1(\boldsymbol{x}) \leq V(t, \boldsymbol{x}) \leq W_2(\boldsymbol{x}) \tag{A.2}$$

$$\frac{\partial V}{\partial t} + \left(\frac{\partial V}{\partial \boldsymbol{x}}\right)^T \boldsymbol{f}(t, \boldsymbol{x}) \leq -\alpha(W_1(\boldsymbol{x})) \tag{A.3}$$

for all $t \geq 0$, $x \in \mathcal{D}$, and some $\alpha \in \mathcal{K}$, with $\mathcal{K}$ being strictly increasing, continuous functions. Then the equilibrium $\boldsymbol{x} = 0$ is UAS if it is Ω-UAS, where $\Omega = \{\boldsymbol{x} \in \mathcal{D} | W_1(\boldsymbol{x}) = 0\}$.

ACKNOWLEDGMENT

Thanks to Henrik Schmidt-Didlaukies, who originally developed the simulator of the Eelume and made it available for use, and to Saray Bakker for help with implementation

REFERENCES

[1] A. Dietrich, C. Ott, and A. Albu-Schäffer, "An overview of null space projections for redundant, torque-controlled robots," *The International Journal of Robotics Research*, vol. 34, no. 11, pp. 1385–1400, 2015.

[2] B. Siciliano and J. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Proc. 5th Int. Conf. on Advanced Robotics*, Pisa, Italy, 1991.

[3] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 3, pp. 398–410, 1997.

[4] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, 1987.

[5] O. Kanoun, F. Lamiraux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Trans. on Robotics*, vol. 27, no. 4, pp. 785–792, 2011.

[6] K. Bouyarmane and A. Kheddar, "On weight-prioritized multitask control of humanoid robots," *IEEE Transactions on Automatic Control*, vol. 63, no. 6, pp. 1632–1647, 2018.

[7] G. Antonelli and S. Chiaverini, "Task-priority redundancy resolution for underwater vehicle-manipulator systems," in *Proc. 1998 IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgium, 1998.

[8] E. A. Basso and K. Y. Pettersen, "Task-priority control of redundant robotic systems using control lyapunov and control barrier function based quadratic programs," in *Proc. IFAC World Congress*, Berlin, Germany, July 2020.

[9] I.-L. G. Borlaug, K. Y. Pettersen, and J. T. Gravdahl, "Combined kinematic and dynamic control of vehicle-manipulator systems," *Mechatronics*, vol. 69, p. 102380, 2020.

[10] N. Hogan, "Impedance control: An approach to manipulation," in *Proc. 1984 American Control Conference*, San Diego, USA, 1984, pp. 304–313.

[11] P. Dai, W. Lu, K. Le, and D. Liu, "Sliding mode impedance control for contact intervention of an I-AUV: Simulation and experimental validation," *Ocean Engineering*, vol. 196, p. 106855, 2020.

[12] P. Cieślak and P. Ridao, "Adaptive admittance control in task-priority framework for contact force control in autonomous underwater floating manipulation," in *Proc. 2018 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Madrid, Spain, 2018, pp. 6646–6651.

[13] A. Dietrich and C. Ott, "Hierarchical impedance-based tracking control of kinematically redundant robots," *IEEE Trans. on Robotics*, vol. 36, no. 1, pp. 204–221, 2020.

[14] H. Sadeghian, V. Luigi, M. Keshmiri, and B. Siciliano, "Task-space control of robot manipulators with null-space compliance," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 493–506, April 2014.

[15] C. Ott, A. Dietrich, and A. Albu-Schäffer, "Prioritized multi-task compliance control of redundant manipulators," *Automatica*, vol. 53, pp. 416–423, 2015.

[16] G. Garofalo and C. Ott, "Hierarchical tracking control with arbitrary task dimensions: Application to trajectory tracking on submanifolds," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6153–6160, October 2020.

[17] G. Garofalo, X. Wu, and C. Ott, "Adaptive passivity-based multi-task tracking control for robotic manipulators," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7129–7136, 2021.

[18] P. From, J. Gravdahl, and K. Pettersen, *Vehicle-Manipulator Systems*. Springer-Verlag London, 2014.

[19] G. Antonelli, *Underwater Robots*, 4th ed. Springer Int. Publishing AG Cham, 2018.

[20] J.-J. Slotine and W. Li, "On the adaptive control of robot manipulators," *Int. J. Robotics Research*, vol. 6, no. 3, pp. 49–59, 1987.

[21] H. Berghuis and H. Nijmeijer, "Passivity approach to controller-observer design for robots," *IEEE Transactions on Automatic Control*, vol. 9, no. 6, pp. 740–754, 1993.

[22] N. Sadegh and R. Horowitz, "Stability and robustness analysis of a class of adaptive controllers for robotic manipulators," *Int. J. Robotics Research*, vol. 9, no. 3, pp. 74–92, 1990.

[23] X. Wu, C. Ott, A. Albu-Schäffer, and A. Dietrich, "Passive decoupled multi-task controller for redundant robots," *TechRxiv. Preprint*, November 2021. [Online]. Available: https://doi.org/10.36227/techrxiv.17025314.v1

[24] H. K. Khalil, *Nonlinear Systems*. 3rd ed. Prentice Hall, 2002.

[25] Z. Wang, Y. Tan, G. Wang, and D. Nešic, "On stability properties of nonlinear time-varying systems by semi-definite time-varying lyapunov candidates," in *Proc. 17th IFAC World Congress*, Seoul, Korea, July 6-11 2008, pp. 1123–1128.

[26] H. M. Schmidt-Didlaukies, A. J. Sørensen, and K. Y. Pettersen, "Modeling of articulated underwater robots for simulation and control," in *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, 2018, pp. 1–7.