

Real-Time Model Predictive Control for Energy Management in Autonomous Underwater Vehicle

Niankai Yang¹, Mohammad Reza Amini¹, Matthew Johnson-Roberson¹ and Jing Sun¹

Abstract—Improving endurance is crucial for extending the spatial and temporal operation range of autonomous underwater vehicles (AUVs). Considering the hardware constraints and the performance requirements, an intelligent energy management system is required to extend the operation range of AUVs. This paper presents a novel model predictive control (MPC) framework for energy-optimal point-to-point motion control of an AUV. In this scheme, the energy management problem of an AUV is reformulated as a surge motion optimization problem in two stages. First, a system-level energy minimization problem is solved by managing the trade-off between the energies required for overcoming the positive buoyancy and surge drag force in static optimization. Next, an MPC with a special cost function formulation is proposed to deal with transients and system dynamics. A switching logic for handling the transition between the static and dynamic stages is incorporated to reduce the computational efforts. Simulation results show that the proposed method is able to achieve near-optimal energy consumption with considerable lower computational complexity.

I. INTRODUCTION

Autonomous underwater vehicles (AUVs) are advancing the state-of-the-art in numerous oceanography and aquatic environmental monitoring applications. With high level of autonomy, AUVs are well-suited for deep water or long-range explorations. Endurance is a key consideration for AUV design and operation. With constraints on the construction cost and other competing design requirements, the on-board energy storage of AUV is often limited, making AUV endurance a challenge and priority for design and operation.

Extensive efforts were made to improve the energy consciousness of AUV control in either the planning stage or the execution stage. At the planning stage, the ocean currents, which can be comparable in magnitude to the average vehicle operational speed, are considered to obtain the optimal trajectories associated with the minimum energy consumption [1]. In [2], a stochastic optimization method integrated with data-driven ocean modeling is presented to generate an energy-optimal trajectory with reference speed and heading. The Pontryagin's maximum principle (PMP) was applied in [3] to obtain the global optimal energy trajectory. However, the intensive computation associated with the above planning algorithms makes their real-time implementation infeasible. Moreover, since the planning is performed offline, the results are sensitive to uncertainties in the AUV model and time-varying disturbances.

Works have also been reported on the design of trajectory tracking controllers for energy consumption reduction at the execution level. In [4], a sliding mode controller was designed with Euler-Lagrange based classical optimal control for minimizing the control efforts and saving energy. State-dependent Riccati equation was applied in [5] for an energy-efficient AUV tracking control. These control strategies assume a given planned trajectory and achieve energy savings by compromising the tracking accuracy.

In this paper, we propose an execution-level controller for a point-to-point motion control of an AUV, while optimizing the vehicle trajectory online by utilizing dynamic vehicle model to improve the energy efficiency of the system. To this end, an innovative model predictive control (MPC) framework with low computational complexity is developed to facilitate real-time implementation. The main contribution of this paper relies on reformulating the energy management problem of an AUV as a surge motion optimization problem, which incorporates the heave-related energy spending as a terminal cost in MPC to capture the “cost-to-go”. In order to reduce the computational efforts, the surge motion optimization is performed in static and dynamic stages via a switching strategy in the proposed MPC.

II. AUV SIMULATION MODEL

In this study, the DROP-Sphere [6], a low cost, 6000m rated AUV designed for optical benthic mapping of the deep sea is used as the virtual testbed for algorithm development.

A. DROP-Sphere Configuration

DROP-Sphere, whose scheme is shown in Fig.1, is propelled by four hub-less bi-directional thrusters powered by DC motors. Two horizontal thrusters are used for the surge and yaw controls, while two vertical thrusters are used for the heave and pitch controls. All the electrical devices (e.g., battery, camera, microprocessor) are mounted inside a transparent sphere located at the middle of the vehicle.

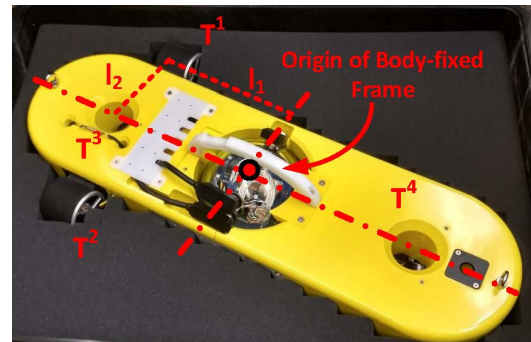


Fig. 1. Schematic of the DROP-Sphere

¹Niankai Yang, Mohammad Reza Amini, Matthew Johnson-Roberson and Jing Sun are with Department of Naval Architecture and Marine Engineering, University of Michigan, Ann Arbor, MI 48109 USA. (e-mail: {ynk, mamini, mattjr, jingsun}@umich.edu)

B. Mathematical Modeling

To develop the equations of motion of an AUV, body-fixed and earth-fixed coordinate frames [7] are employed. Then the velocity, position, orientation, force and moment components (shown in Fig. 2) are defined as follows:

$$\nu = [u \ v \ w \ p \ q \ r]^T, \quad (1)$$

$$\eta = [x \ y \ z \ \phi \ \theta \ \psi]^T, \quad (2)$$

$$\tau = [X \ Y \ Z \ K \ M \ N]^T, \quad (3)$$

where, ν , η , and τ are the vectors of velocities, positions and orientations, and external forces and moments respectively.

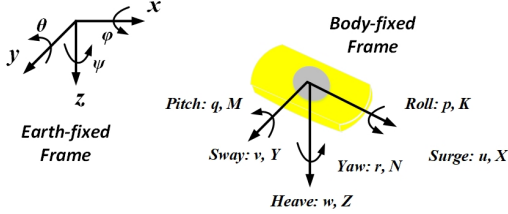


Fig. 2. Reference Frames and Notations

Based on the defined reference frames and notations, a general description of the nonlinear coupled AUV model is derived through the Newton-Euler equations of motion for a rigid body [8]. A compact form of the model is given as:

$$\begin{cases} M_t \dot{\nu} + F_c(\nu) + F_h(\nu)\nu + F_g(\eta) = \tau_c, \\ \dot{\eta} = J(\eta)\nu, \end{cases} \quad (4)$$

where, M_t is the total mass, $F_h(\nu)$ is the hydrodynamic damping matrix, τ_c is the control inputs. The $F_c(\nu)$, $F_g(\eta)$ and $J(\eta)$ detailed in [9] are the Coriolis and centripetal force, the hydrostatic force, and the coordinate transformation between the body-fixed and earth-fixed frames.

The total mass consists of the rigid body mass and the added mass. For the sake of simplicity and without consequential loss of accuracy, only diagonal terms in the added mass matrix (M_a) are considered:

$$M_a = -\text{diag}(X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}), \quad (5)$$

where, $X_{\dot{u}}$, $Y_{\dot{v}}$, $Z_{\dot{w}}$, $K_{\dot{p}}$, $M_{\dot{q}}$, $N_{\dot{r}}$ are the added mass coefficients. Similarly, only diagonal and quadratic drag terms in hydrodynamic damping matrix are considered:

$$F_h(\nu) = -\text{diag}(X_{|u|u}, Y_{|v|v}, Z_{|w|w}, K_{|p|p}, M_{|q|q}, N_{|r|r}) \cdot |\nu|, \quad (6)$$

where, $X_{|u|u}$, $Y_{|v|v}$, $Z_{|w|w}$, $K_{|p|p}$, $M_{|q|q}$, $N_{|r|r}$ are the quadratic drag coefficients.

The control input vector τ_c consists of six components representing the control forces or moments in each DOF in the body-fixed frame. By combining the input transformation matrix with the thruster inputs, τ_c can be expressed as:

$$\begin{aligned} \tau_c &= [\tau_X \ \tau_Y \ \tau_Z \ \tau_K \ \tau_M \ \tau_N]^T \\ &= [T^1 + T^2 \ 0 \ T^3 + T^4 \ 0 \ l_1(T^3 - T^4) \ l_2(T^1 - T^2)]^T, \end{aligned} \quad (7)$$

where, as shown in Fig. 1, T^i is the input force of the i^{th} thruster. l_1 is the distance between the vertical thrusters and the midship. l_2 is the distance from the horizontal thrusters to the center line.

The thruster inputs are converted from force to power in order to facilitate the energy consumption analysis. The power consumption of unit input force is calculated according to the momentum conservation theory as [10]:

$$P(T^i) = \sqrt{\frac{1}{2\pi\rho}} \cdot \frac{(T^i)^{1.5}}{R} = C_p(T^i)^{1.5}, \quad (8)$$

in which, $P(T^i)$ is the power consumption of the i^{th} thruster, ρ is the water density, R is the radius of thruster, and C_p is defined as the power conversion ratio. The numerical values for the vehicle parameters are provided in Appendix I.

III. SURGE MOTION OPTIMIZATION FORMULATION

A. Energy Management Problem Formulation

In this paper, we consider energy-optimal maneuvering of the DROP-Sphere between two horizontal waypoints in an obstacle-free underwater area with no ocean currents. The vehicle is located at the initial position (x_0) heading towards the final position (x_f) with zero initial velocity in sway, heave, roll, pitch and yaw. While the controller is designed to drive the vehicle from x_0 to x_f , the energy consumed by the thruster input sequence $\{T_k^i\}$ is defined as:

$$J(x_f - x_0, u_0, \{T_k^i\}) = \sum_{k=0}^{n-1} \left(\sum_{i=1}^4 P(T_k^i) \cdot \frac{t_{travel}}{n} \right), \quad (9)$$

with $x_f - x_0$ being the total distance, k the index for step time, n the number of samples, and t_{travel} the travel time.

The minimization of J in (9) is subject to (i) 6 DOF system dynamics described in (4), and (ii) the state and input constraints given by the inequalities:

$$C_{min} \leq C_k \leq C_{max}, \quad (10)$$

where, $(\cdot)_{min}$ and $(\cdot)_{max}$ are the minimum and maximum of the corresponding variables, $C_k = [T_k^i, y_k, z_k, \phi_k, \theta_k, \psi_k]^T$.

B. Optimal Solution of the Energy Management Problem

Treated as a trajectory optimization problem, the problem in (9) can be solved with direct collocation (DC) [11] or other numerical methods to obtain the global optimal solution. In this study, we consider the maneuvering from $x_0 = 0$ to $x_f = 10$ with $u_0 = 0$. The state equation is discretized into 300 segments and approximated with the trapezoid rule to guarantee the optimality and robustness. The constraints are set as $C_{max} = -C_{min} = [7.84, 0.01, 0.005, 0.2, 0.01, 0.01]^T$. The units for the variables are N for thrusts, m for positions and rad for orientations.

As illustrated by the resulted position and orientation traces shown in Fig. 3, the optimal solution achieves point-to-point motion control while enforcing the constraints. However, DC method suffers from two issues. The open-loop nature of the solution makes it sensitive to modeling errors. In addition, their intensive computation renders the real-time implementation impractical. For example, solving (9) with DC method for 74.04 s of simulation takes more than 3 hours on a 2.9 GHz Intel Core i5 processor with 8GB RAM.

C. Optimization Problem Reformulation

To gain insights into the optimal operation of the DROP-Sphere, an in-depth analysis of the optimal solution is conducted. The total energy consumption (69.08 J) resulted from the DC method is distributed for surge (22.08 J), heave (46.28 J), pitch (0.16 J) and yaw (0.01 J) controls. Note that the energy used for controlling yaw (0.01 %) and pitch (0.23 %) are negligible compared to that of surge (32.92 %)

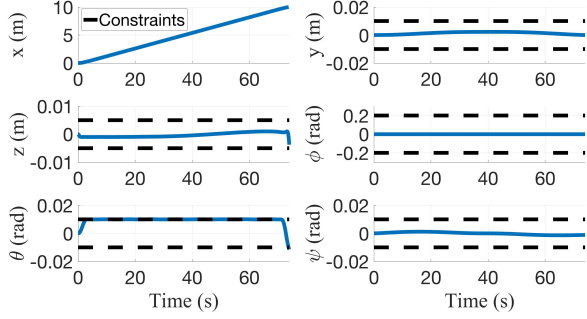


Fig. 3. Vehicle Trajectories from Direct Collocation Method

and heave (66.83%) controls. Given the constraints in (10), then the yaw and pitch angles will be nearly zero during the whole trajectory. Thus, under this point-to-point motion control problem formulation, the energy saving potential from yaw and pitch controls is minimum.

Moreover, it can be observed from Fig. 4 that the heave power stays constant for most of the simulation time. By inspecting the heave dynamics in (4), this constant value equals the power used for balancing the difference between the buoyancy and weight of the vehicle, i.e., the positive buoyancy. Given the pitch moment is approximately zero, the heave power can be calculated as:

$$P^{PB} = \frac{\sqrt{2}}{2} C_p (B - W)^{1.5}, \quad (11)$$

where, W and B are the weight and buoyancy. Since the heave power is almost constant, the total heave energy becomes a linear function of the traveling time. This means that the heave energy is a reciprocal function of the surge speed, if the total distance is fixed. Thus, the energy saving could be substantial by increasing the surge velocity. On the other hand, higher surge speed leads to larger surge drag force ($X_{|u|u}|u|u$), which ultimately results in higher surge energy consumption. Thus, an optimization using the surge speed as the key variable can properly capture the sensitivity and trade-off of energy consumption to vehicle operation.

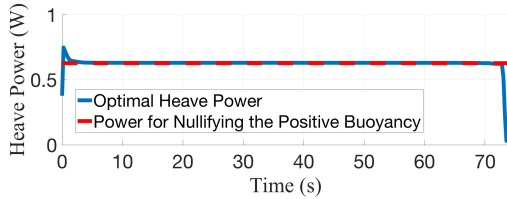


Fig. 4. Heave Power Time History of the Optimal Solution

The analysis of the optimal solution leads to a decentralized control architecture where the yaw, pitch, and heave are independently controlled by three PID controllers and the energy management problem in (9) is reformulated as a surge motion optimization problem. This results in a much simpler problem formulation compared to the original energy management problem. The surge motion optimization aims at finding the total horizontal thruster input sequence $\{T_k^{total}\}$ to minimize the following cost function:

$$J_{surge}(x_f - x_0, u_0, \{T_k^{total}\}) = \sum_{k=0}^{n-1} (2 \cdot P(\frac{1}{2} T_k^{total}) + P^{PB}) \cdot \frac{t_{travel}}{n}, \quad (12)$$

where, T_k^{total} equals the combination of two horizontal thruster forces, which are same in magnitude and direction when the yaw moment is nearly zero. The optimization of J_{surge} is subject to (i) input and state (x_f , x_0 , u_0) constraints, and (ii) 2 DOF dynamically decoupled surge dynamics (i.e., the accelerations from other DOF are considered as zero, while the velocities and orientations are assumed to be constant over the prediction horizon) described as:

$$(m - X_{\ddot{u}})\ddot{u} + m(wq - vr + z_g pr) + (W - B)s\dot{\theta} = -X_{|u|u}|u|u + T^{total}, \quad (13)$$

$$\dot{x} = (c\psi c\theta)u + (c\psi s\theta s\phi - s\psi c\phi)v + (s\psi s\phi + c\psi s\theta c\phi)w, \quad (14)$$

where, z_g is the center of gravity in the heave direction, c and s are $\cos(\cdot)$ and $\sin(\cdot)$ respectively.

IV. STATIC SURGE MOTION OPTIMIZATION

One can approach the optimization problem defined in (12) by tracking an optimal surge velocity setpoint derived from solving a static optimization problem (i.e., assuming constant surge velocity). Note that from the optimal solution, variations in the responses of the state variables other than surge velocity and x position is minor. If we further assume that the surge speed is constant, then the total horizontal thrust (T^{total}) will be equal to the surge drag force ($X_{|u|u}|u|u$) in (13). According to the heave power expression in (11), the total energy consumption can be described as:

$$J_{static}(x_f - x_0, u_0) = (x_f - x_0) \cdot EPD(u_0), \quad (15)$$

where, EPD is *energy cost per distance* defined as:

$$EPD(u) = \frac{\sqrt{2}}{2} C_p X_{|u|u}^{1.5} u^2 + \frac{P^{PB}}{u}. \quad (16)$$

The absolute operator on the surge velocity in $X_{|u|u}|u|u$ is dropped, as the negative surge velocity is not considered in this study. Based on (15), the minimization of total energy will be equivalent to the minimization of EPD in (16) for given $x_f - x_0$. Since EPD consists of a reciprocal function and a quadratic function of surge speed, there is a static optimal surge velocity (u_{static}^*) minimizing EPD as well as the total energy consumption:

$$u_{static}^* = \arg \min_u \{EPD(u)\}, \quad (17)$$

which represents the best trade-off between the energies for overcoming the positive buoyancy and the surge drag force.

To track the calculated u_{static}^* , a setpoint tracking MPC (T-MPC) is designed. The T-MPC calculates the total horizontal thruster input sequence $\{T_{k|t}^{total}\}$ over a prediction horizon N to minimize the following cost function:

$$J_T(u_{static}^*, u_t, \{T_{k|t}^{total}\}) = \sum_{k=0}^{N-1} (u_{static}^* - u_{k+1|t})^2, \quad (18)$$

where, $(\cdot)_{k+1|t}$ is the $k+1$ -step ahead prediction made at time t . The optimization in (18) is subject to the decoupled surge dynamic model in (13) and (14), and the input constraints.

The T-MPC results in a sub-optimal performance in terms of energy consumption. For this case study, it results in 5.11% additional energy cost compared to the optimal DC solution (i.e., 72.61J vs. 69.08J). A comparison of surge

velocity trajectory and total horizontal input from DC and T-MPC are shown in Fig. 5. It can be seen from Fig. 5 that for most of the time, solution from T-MPC is close to that of DC with the same surge velocity, which verifies that tracking a statically optimized velocity setpoint can lead to “almost” optimal solution. However, during the acceleration and deceleration phases, the response of T-MPC differs from DC substantially. This is expected as T-MPC aims at reducing the tracking error regardless of the energy consumption. This motivates the dynamic surge motion optimization approach proposed in the next section to take the energy consumption into consideration during the dynamic stage.

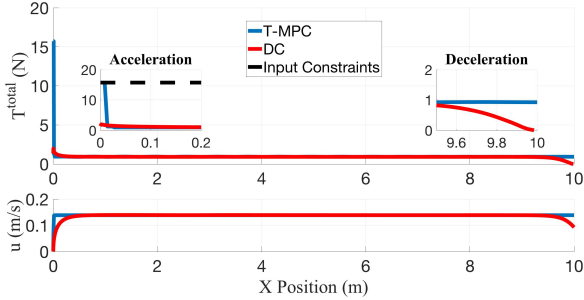


Fig. 5. Performance Comparison of T-MPC and DC

V. DYNAMIC SURGE MOTION OPTIMIZATION

In order to incorporate energy-consciousness into the MPC, one can consider the following cost function:

$$J_L(x_t, u_t, \{T_{k|t}^{total}\}) = \sum_{k=0}^{N-1} (2 \cdot P(\frac{1}{2}T_{k|t}^{total}) + P^{PB})\Delta t, \quad (19)$$

where, Δt is the sampling time. The policy to optimize (19), however, will set the control input to zero for minimizing the energy consumption, unless the terminal position (x_f) is included in the MPC formulation as a state constraint and a long prediction horizon is used. With a long prediction horizon, the MPC becomes almost identical to those computationally demanding global optimization methods.

To overcome this problem, we propose to add a terminal cost term (J_K) to J_L in (19) to reflect the “cost-to-go” and ensure that the MPC controller takes into account both the energy saving and the destination reaching. This terminal cost J_K should approximate the “cost-to-go” and satisfy the following two requirements: (i) J_K is an increasing function of the distance between x_f and the x position at the end of prediction horizon ($x_{N|t}$). (ii) J_K provides an estimation of the energy consumption for reaching x_f from $x_{N|t}$.

Recalling the discussion in Section. IV, we propose the following function as the terminal cost:

$$J_K = \frac{x_f - x_{N|t}}{u_{N|t}} \cdot (P^{PB} + \frac{\sqrt{2}}{2}C_p X_{u|u}^{1.5} u_{N|t}^3), \quad (20)$$

where, $u_{N|t}$ is surge velocity at the end of the prediction horizon. Then the overall cost function of the modified MPC called the energy-optimal MPC (EO-MPC) is described as:

$$J_{EO}(x_f - x_t, u_t, \{T_{k|t}^{total}\}) = J_L + J_K, \quad (21)$$

where J_L is defined in (19). The optimization problem of (21) is subject to the same constraints as T-MPC.

According to the Bellman’s Principle of Optimality [12], the EO-MPC would yield the optimal solution if J_K were

indeed the “cost-to-go”, namely, $J_K(x_f - x_{N|t}, u_{N|t}) = J_{DC}^*(x_f - x_{N|t}, u_{N|t}, \{T_k^i\})$, where J_{DC}^* is the global optimal solution from (9). Extensive analysis has been carried out to evaluate the sensitivity of J_K in estimating J_{DC}^* with respect to different initial surge velocities at different distances to x_f . It can be seen from Fig. 6 that if the distance to x_f is large, J_K is close to J_{DC}^* only when $u_{N|t}$ is near u_{static}^* . Thus, if the prediction horizon is tuned to ensure that (19) is used for approximating the energy consumption in cost function during the region where the vehicle accelerates from u_t to u_{static}^* , then a near-optimal energy consumption can be obtained with EO-MPC.

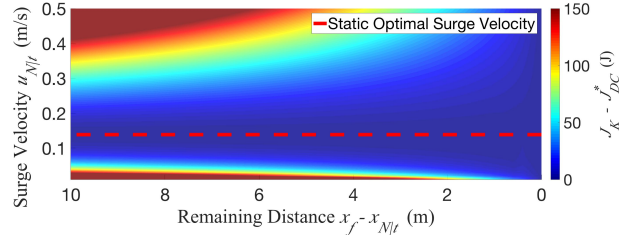


Fig. 6. $J_k - J_{DC}^*$ for Different Initial Conditions

Fig. 7 shows the effect of the prediction horizon on energy consumption and the maximum CPU time for solving the optimization problem per iteration. It can be seen that as the length of the MPC prediction horizon increases, the energy consumption ultimately decreases and converges to 69.84J, which is close to the energy consumption from the global optimal solution (69.08J). On the other hand, the CPU time increases as the prediction horizon becomes longer. Thus, in order to avoid high computational complexity and achieve near-optimal energy consumption, the prediction horizon can be selected as the minimum value that ensures the performance convergence [13]. Additionally, the maximum CPU time per iteration should also be smaller than the sampling time to guarantee the real-time execution of the MPC. For this case study, based on Fig. 7, the prediction horizon is chosen as $N = 15$. It should be noted that because of the randomness associated with solving the optimization problem, the CPU time is calculated by taking average of the results from running the problem repeatedly for 10 times.

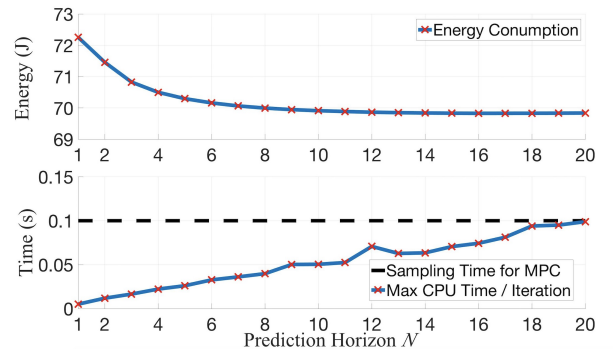


Fig. 7. Prediction Horizon Sensitivity Analysis

VI. REAL-TIME ENERGY-OPTIMAL MPC

With the proposed EO-MPC in Sec. V, the controller is able to achieve real-time near-optimal point-to-point motion control. However, as it was indicated from the performance

comparison between T-MPC and DC, the benefit of conducting dynamic optimization in steady state is minimum. To further simplify the computational complexity of EO-MPC, a switching MPC algorithm is developed as the real-time energy-optimal MPC (RTEO-MPC). The RTEO-MPC takes different actions during the static and dynamic stages of the point-to-point motion control. As shown in Algorithm (1), at the beginning where the initial speed is different than the static optimal surge velocity (u_{static}^*) and at the end of the trajectory where the vehicle needs to decelerate in order to save energy, the RTEO-MPC has the same structure of the EO-MPC. During the cruise operation where the vehicle operates close to the u_{static}^* , the RTEO-MPC switches to a simple logic to keep the vehicle at a constant surge speed. The overall schematic of the real-time energy-optimal MPC (RTEO-MPC) along with the PIDs for depth and steering controls are illustrated in Fig. 8.

Algorithm 1 Switching Strategy for RTEO-MPC

- 1: Given the vehicle configuration, x_f , compute u_{static}^* ;
- 2: Set u_{low}^{switch} , u_{high}^{switch} , x^{switch} ;
- 3: **If** $x_t < x^{switch}$
- 4: **If** $u_0 < u_{static}^*$
- 5: **If** $u_t < u_{low}^{switch}$ or $u_{t-1} < u_t$
- 6: Compute T_t^{total} using EO-MPC;
- 7: **Else**
- 8: Set T_t^{total} same as T_{t-1}^{total} ;
- 9: **Else**
- 10: **If** $u_t > u_{high}^{switch}$ or $T_{t-2}^{total} < T_{t-1}^{total}$
- 11: Compute T_t^{total} using EO-MPC
- 12: **Else**
- 13: Set T_t^{total} same as T_{t-1}^{total} ;
- 14: **Else**
- 15: Compute T_t^{total} using EO-MPC;

Note: t is the current time. u_{low}^{switch} and u_{high}^{switch} are the low and high bounds around the u_{static}^* . x^{switch} is the starting X position after which the vehicle is near the destination.

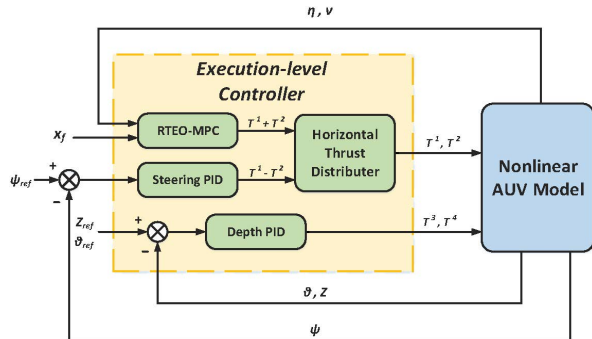


Fig. 8. Architecture of the Control System

VII. SIMULATION RESULTS AND ANALYSIS

This section presents the simulation results verifying the effectiveness of the proposed RTEO-MPC. In this case study, x_0 , u_0 , and x_f are chosen as 0, 0, and 10. The thruster input constraint is considered as $-15.72 N \leq T^{total} \leq 15.72 N$. The sampling time is set as 0.1s. The MPC is implemented in MATLAB/Simulink on the full order model, as the virtual testbed, which provides full state feedback to the controllers.

TABLE I
PERFORMANCE COMPARISON

Control Strategy	Travel Time (s)	Energy Consumption (J)	Loss (%)
DC	74.04	69.08	—
T-MPC	72.20	72.61	5.11
EO-MPC	75.15	69.84	1.10
RTEO-MPC	74.70	69.83	1.09

The energy consumption and the traveling time resulted from using the RTEO-MPC are listed in Table (I). The performance of three other algorithms: (a) direct collocation (DC), (b) MPC for tracking velocity setpoint derived from static optimization (T-MPC), and (c) energy-optimal MPC (EO-MPC) are included for comparison. As indicated in Table (I), the performance of the proposed RTEO-MPC is close to the optimal baseline (DC) with 1.09% more energy consumption. This difference is caused by assuming constant heave power in the cost function. Besides, compared to T-MPC, a 3.83% reduction in energy cost is achieved with the RTEO-MPC. The RTEO-MPC improves the energy efficiency by leveraging the vehicle dynamics and optimizing the surge velocity trajectory during the dynamic stage, while, in the static stage, it guarantees the surge velocity to be close to the static optimal surge velocity (u_{static}^*) as shown in Fig. 9. The time history of positions and orientations resulted from RTEO-MPC is illustrated in Fig. 10. It can be seen from Fig. 10, all the constraints enforced on the original energy management problem are satisfied by using the RTEO-MPC.

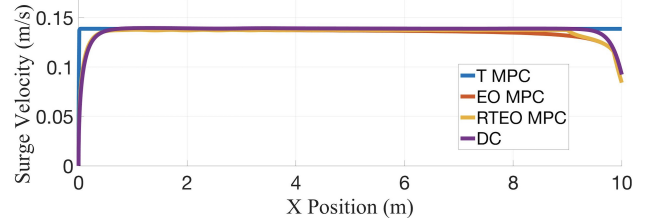


Fig. 9. Surge Velocity Trajectory Comparison

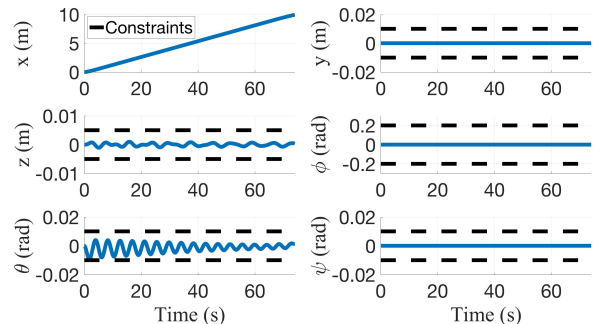


Fig. 10. Vehicle Trajectories from the RTEO-MPC

A further comparison of the computation time using two energy-optimal MPCs (EO-MPC and RTEO-MPC) and the DC method is presented in Table (II). It can be seen that by using the switching strategy, the RTEO-MPC reduces the average computation time by 74.59%, without any deterioration in the energy efficiency compared to the EO-MPC. Moreover, compared to the DC method, the RTEO-MPC reduces the total computation time by 99.98%.

TABLE II
COMPUTATION TIME COMPARISON

Control Strategy	Average CPU Time / Sampling Time (s)	Total CPU Time for the Simulation of the Trip (s)
EO-MPC	0.0122	9.1824
RTEO-MPC	0.0031	2.3503
DC	–	11267.8

To illustrate the robustness of the RTEO-MPC against different initial conditions, the energy consumption obtained from the RTEO-MPC (E_{RTEO}^*) and the T-MPC (E_T^*), with different initial positions and velocities, are compared to that of the DC method (J_{DC}^*) respectively, and the results are shown in Fig. 11. The ranges of initial position and velocity are set as $x_0 \in [0, 10]$ and $u_0 \in [0, 0.5]$. It can be seen from Fig. 11 that the performance of the T-MPC deteriorates as the difference of initial velocity and the static optimal surge velocity increases, while the proposed RTEO-MPC is able to provide a near-optimal energy consumption under different initial conditions by combining dynamic and static optimizations via a switching logic. Higher energy consumption occurs for both the RTEO-MPC and the T-MPC when the vehicle starts very close to the target destination with low velocity, which is caused by neglecting the coupling effect of the heave and surge dynamics in the surge motion optimization problem formulation. However, this scenario is less likely to happen in an AUV deployment.

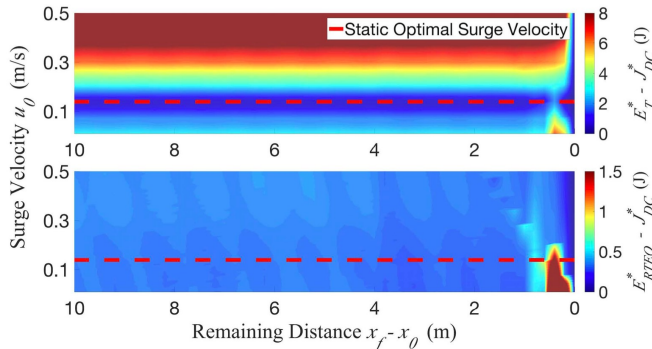


Fig. 11. $E_T^* - J_{DC}^*$ and $E_{RTEO}^* - J_{DC}^*$ for Different Initial Conditions

VIII. CONCLUSION

In this paper, an innovative energy-optimal model predictive controller (MPC) was developed for real-time point-to-point motion control and energy management optimization of an AUV. To reduce the computational efforts and allow for real-time implementation of the proposed MPC, the overall AUV controller was designed and integrated based on a decentralized control architecture with the proposed MPC combined with several PIDs. The underlying physics of the vehicle motion and energy consumption is exploited by formulating the MPC cost function in terms of the total energy cost for transversing the remaining distance. Exploring the "static optimal" that achieves the best trade-off between energies for surge and heave controls is achieved, a switching logic was incorporated into the developed RTEO-MPC to switch the control law between static and dynamic stages of the surge motion for further reduction in the computation time. Simulation results verified that the proposed method

allows for the real-time trajectory optimization with near-optimal energy consumption under different initial conditions. For a case study presented, the RTEO-MPC showed a 3.83% improvement in energy consumption compared to the MPC for tracking static optimal surge velocity setpoint (T-MPC), and 99.98% reduction in total computation time compared to the global optimization method (DC).

Future works will be focusing on generalizing this framework to (i) complicated maneuvering operations (e.g., steering and diving), and (ii) realistic environmental conditions (e.g., ocean currents and obstacles). Meanwhile, theoretical results about the recursive feasibility and optimality resulted from the proposed terminal cost will be investigated for the point-to-point navigation scenario.

ACKNOWLEDGMENT

The authors thank Dr. Corina Barbalata, and Mr. Eduardo Iscar Rüland from DROP lab at the University of Michigan for providing details of the AUV model and their technical comments during this study.

REFERENCES

- [1] D. N. Subramani, T. Lolla, P. J. Haley, and P. F. Lermusiaux, "A stochastic optimization method for energy-based path planning," in *Dynamic Data-Driven Environmental Systems Science*. Springer, 2015, pp. 347–358.
- [2] D. N. Subramani, P. J. Haley, and P. F. Lermusiaux, "Energy-optimal path planning in the coastal ocean," *Journal of Geophysical Research: Oceans*, vol. 122, no. 5, pp. 3981–4003, 2017.
- [3] M. Chyba, T. Haberkorn, S. Singh, R. Smith, and S. Choi, "Increasing underwater vehicle autonomy by reducing energy consumption," *Ocean Engineering*, vol. 36, no. 1, pp. 62–73, 2009.
- [4] M. Sarkar, S. Nandy, S. Vadali, S. Roy, and S. Shome, "Modelling and simulation of a robust energy efficient auv controller," *Mathematics and Computers in Simulation*, vol. 121, pp. 34–47, 2016.
- [5] B. Geranmehr and S. R. Nekoo, "Nonlinear suboptimal control of fully coupled non-affine six-dof autonomous underwater vehicle using the state-dependent riccati equation," *Ocean Engineering*, vol. 96, pp. 248–257, 2015.
- [6] E. Iscar, C. Barbalata, N. Goumas, and M. Johnson-Roberson, "Towards low cost, deep water auv optical mapping," in *IEEE/MTS Oceans*, 2018.
- [7] S. Wadoo and P. Kachroo, *Autonomous underwater vehicles: modeling, control design and simulation*. CRC Press, 2017.
- [8] J. Yuh, "Modeling and control of underwater robotic vehicles," *IEEE Transactions on Systems, man, and Cybernetics*, vol. 20, no. 6, pp. 1475–1483, 1990.
- [9] T. T. J. Prestero, "Verification of a six-degree of freedom simulation model for the remus autonomous underwater vehicle," Ph.D. dissertation, Massachusetts institute of technology, 2001.
- [10] S. Baluta, "How much power is needed to hovering," <http://www.starlino.com/power2thrust.html>, accessed February 12, 2018.
- [11] O. Von Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Annals of operations research*, vol. 37, no. 1, pp. 357–373, 1992.
- [12] R. Bellman, *Dynamic programming*. Courier Corporation, 2013.
- [13] W. Wojcniński, J. Gudaz, T. Blevins, and A. Mehta, "Practical approach to tuning mpc," *ISA transactions*, vol. 42, no. 1, pp. 149–162, 2003.

APPENDIX I

SPHERE VEHICLE PARAMETERS

$W = 200.116 \text{ N}$	$B = 201.586 \text{ N}$	$m = 20.42 \text{ kg}$
$I_{xx} = 0.1205 \text{ kg m}^2$	$I_{yy} = 0.9431 \text{ kg m}^2$	$I_{zz} = 1.0061 \text{ kg m}^2$
$z_g = 0.0018 \text{ m}$	$l_1 = 0.1694 \text{ m}$	$l_2 = 0.2794 \text{ m}$
$R = 0.025 \text{ m}$	$X_u = -2.042 \text{ kg}$	$Y_v = -32.2013 \text{ kg}$
$Z_w = -32.2013 \text{ kg}$	$K_p = -0.0805 \text{ kg}$	$M_q = -2.6834 \text{ kg}$
$N_r = -2.6834 \text{ kg}$	$X_{u u} = 48.17 \text{ kg/m}$	$Y_{v v} = 4.11 \text{ kg/m}$
$Z_{w w} = 4.11 \text{ kg/m}$	$K_{p p} = 48.17 \text{ kg/m}$	$M_{q q} = 4.11 \text{ kg/m}$
$N_{r r} = 4.11 \text{ kg/m}$	$\rho = 1.025 \text{ kg/m}^3$	