Efficient Optimal Constrained Control Allocation via Multi-parametric Programming

Tor A. Johansen*, Thor I. Fossen and Petter Tøndel

Department of Engineering Cybernetics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway.

Abstract

Constrained control allocation is studied, and it is shown how an explicit piecewise linear representation of the optimal solution can be computed numerically using multi-parametric quadratic programming. It is demonstrated on several over-actuated F18 aircraft control configurations that the present optimal approach gives significant improvements in real-time computational efficiency compared to sub-optimal linear and piecewise linear explicit approaches from the literature. The applicability of the method is further demonstrated using over-actuated marine vessel dynamic position experiments on a scale model in a basin.

I. Introduction

Control allocation is an important part of ship control systems [1], [2], flight control systems [3], [4] and other over-actuated mechanical control applications, e.g. [5]. The control allocation module will send control signals to the individual actuators (such as thrusters, rudders and propellers in marine vessels, or ailerons, elevators, flaps, rudders and vector thrust devices in aircrafts) in order to produce the required forces and moments (hereafter denoted generalized forces) commanded from a higher level control system or pilot during manual operation.

Such over-actuated control allocation problems are naturally formulated as optimization problems as one usually wants to take advantage of all available degrees of freedom in order to minimize power consumption, drag, tear/wear and other costs related to the use of control, subject to constraints such as actuator position limitations, e.g. [6], [7], [3]. In general, this leads to a constrained optimization problem that is hard to solve using state-of-the-art iterative numerical optimization software at a high sampling rate in a safety-critical real-time system with limiting processing capacity and high demands for software reliability. Still, real-time iterative optimization solutions have been proposed [10], [11], [7], [12], [13], also for some large-scale problems [5], although their worst case computational complexity may be disadvantageous and software verification is a complicated issue. Many practical solutions has therefore focused on simplified optimization problems where simple explicit solutions can be found and implemented efficiently by combining simple matrix computations, logic and filtering [6], [1], [8], [9].

Here we consider over-actuated systems where each of the individual linear actuators produces a bounded force in a fixed direction. In [3], [4], [14] the so-called "direct control allocation" method is suggested, based on pre-computing a polyhedral representation of the attainable moment set, the set of generalized forces that can be attained using feasible controls. The advantage of this approach is that a feasible control allocation can be found if it exists. The main problems of the associated real-time algorithms is that the worst-case computational complexity may be large due to the combinatorial nature of the problem, and that the solution may be sub-optimal. Some modifications and extensions are given in [15], [16], [17], allowing the computational complexity to be reduced at the cost of further sub-optimality via tailored generalized inverses, introducing approximations, and/or reducing the number of constraint combinations being considered. Here, we propose a computationally more efficient approach to exact real-time computation of the optimal solution to a class of control allocation problems that includes the one posed in [3], [4]. We show that constrained control allocation problems leads to a linearly constrained multi-parametric quadratic program (mp-QP). This means that the QP is parameterized by problem data such as the commanded generalized force and possibly also constraint limits and criterion parameters. Using recently developed mp-QP solvers [18], [19], [20], [21] the exact solution

*Corresponding author: Tor.Arne.Johansen@itk.ntnu.no

to such problems can be pre-computed off-line in the explicit form of a piecewise linear (PWL) function of the problem parameters, defined on a polyhedral partition of the space of generalized forces and other parameters. In the real-time implementation the optimal constrained control allocation can thus be computed via the evaluation of a PWL function. These computations can be organized as a binary tree search, [22], [23], with worst-case computational complexity that is logarithmic in the number of polyhedral regions. The suggested approach guarantees that an optimal feasible control allocation is found, if it exists. This is also a property shared by the direct approach [3], [4], but in addition the present approach is optimal in the sense that the cost of control is minimized, which leads to reduced drag and power consumption. The suggested approach provides a significantly more efficient algorithm for implementing the optimal solution than the null-space intersection method [24], which suffers from very high worst-case computational complexity because it must compare in real time a larger combinatorial number of possible combinations of active constraints (saturated actuators) to determine the optimal control allocation.

This paper is organized as follows. In section II the control allocation problem is introduced, and formulated as a multi-parametric program in section III. The method is illustrated on some aircraft control allocation problems from the literature in section IV, and ocean vessel dynamic positioning control allocation experiments using a scale model in a basin in section V. Some conclusions are given in section VI, while the multi-parametric programming approach is briefly described in appendix A, and a real-time search algorithm based on a binary search tree is described in appendix B.

II. The control allocation problem

Let the commanded forces in (x, y, z) be denoted (τ_x, τ_y, τ_z), and the commanded moments in roll, pitch and yaw be denoted $(\tau_\phi, \tau_\theta, \tau_\psi)$. These are stacked in a vector of commanded generalized forces $\tau = (\tau_x, \tau_y, \tau_z, \tau_\phi, \tau_\theta, \tau_\psi)^T$. Assume the system is equipped with N linear actuators with control inputs u_i. If each actuator is characterized by a monotonous nonlinearity, it is implicitly assumed that this nonlinearity is inverted. The kinematics then leads to a relationship between the controls $u = (u_1, u_2, ..., u_N)^T$, orientations $\alpha = (\alpha_1, \alpha_2, ..., \alpha_N)^T$ of the actuators and the generalized forces τ on the following form [2]

$$B(\alpha)u \quad = \quad \tau \tag{1}$$

In many control allocation applications not all 6 components of τ are specified. For example, in aerospace applications one is often only concerned about the three body-axis moments [4], while in dynamic position applications involving surface ocean vessels one is usually concerned only with the 3 components $\tau = (\tau_x, \tau_y, \tau_\psi)$, [2]. If u_i is the force provided by the i-th actuator, $B(\alpha)$ is a $3 \times N$-dimensional matrix, where usually $N \geq 3$ and the i-th column has the general form

$$B_i(\alpha) \quad = \quad \begin{pmatrix} \cos(\alpha_i) \\ \sin(\alpha_i) \\ -\ell_{yi}\cos(\alpha_i) + \ell_{xi}\sin(\alpha_i) \end{pmatrix} \tag{2}$$

The point (ℓ_{xi}, ℓ_{yi}) defines the point where the force produced by the i-th actuator attacks, in a coordinate system with origin in the center of gravity and positive z-axis downwards.

In the present work, it is assumed the angles α are fixed such that (1) defines a set of linear equations, and we write B instead of $B(\alpha)$, hereafter. Neglecting any constraints, the common solution to this problem is the generalized inverse, defined as $B^+ = W^{-1}B^T(BW^{-1}B^T)^{-1}$ assuming the configuration is non-singular such that B has full rank, [1], [3], [4]:

$$u \quad = \quad B^+\tau \tag{3}$$

which solves the least-squares problem

$$\min_u u^T W u \quad \text{subject to} \quad Bu = \tau \tag{4}$$

and $W > 0$ is a weighting matrix. The most important feature of this approach is that it admits an explicit solution (3) that is computationally efficient and easily implemented. In order to improve robustness near singular configurations, some modifications are suggested in [1], [9]. It is, however, of interest to consider more advanced optimization formulations that allows more general cost indices and in particular considers the presence of constraints on u, as this will *in general* improve the performance, [3], [4], [14].

III. Optimization formulation and solution

In this section we suggest an optimization formulation of the control allocation problem. Consider the following optimization problem

$$\min_{u,s,\tilde{u}} \left(s^T Q s + u^T W u + \beta \tilde{u} \right) \tag{5}$$

subject to

$$Bu = \tau + s \tag{6}$$

$$u_{min} \leq u \leq u_{max} \tag{7}$$

$$-\tilde{u} \leq u_1, u_2, ..., u_N \leq \tilde{u} \tag{8}$$

The slack variables s introduced in the first term ensures that constraints (7) on u are fulfilled, if necessary by allowing the resulting generalized force Bu to deviate from its specification τ. The second term of the criterion corresponds to the least-squares criterion (4), while the third term minimizes the largest force among the actuators, due to (8). The parameter $\beta \geq 0$ controls the relative weighting on these two criteria, allowing tradeoffs between average and worst-case use of control to be addressed.

Letting $z = (u, s, \tilde{u})^T$ and $x = (\tau, u_{min}, u_{max}, \beta)^T$ it is straightforward to see that the above optimization problem can be reformulated as follows:

$$\min_{z} \left(z^T H z + z^T F x \right) \tag{9}$$

subject to

$$G_1 z = S_1 x \tag{10}$$

$$G_2 z \leq S_2 x \tag{11}$$

where H, F, G_1, G_2, S_1, S_2 are matrices. Since $W > 0$ implies $H \geq 0$, this defines a convex quadratic program in z parameterized by x. It has recently been found that the solution to such problems is a continuous PWL function $z^*(x)$ defined on an polyhedral partition of any polyhedral domain in the parameter-space [20]. Moreover, an exact representation of this PWL function on any subset of the parameter space can be pre-computed off-line using multi-parametric quadratic programming algorithms [19], [20]. Consequently, it is not necessary to solve the QP (5) - (8) in real time for the current value of τ (and u_{min}, u_{max} and β, if they can change). It suffices to evaluate the known PWL function $z^*(x)$.

A brief overview of the mp-QP algorithm, see [18], [19], is given in Appendix A, and an algorithm for efficient evaluation of such PWL functions using a binary search tree representation is outlined in Appendix B, see [22], [23]. It is worthwhile remarking that in the above formulation both the weighting parameter β and the constraint limits u_{min} and u_{max} are included in the parameter vector x. Considerable reduction of complexity is achieved if these are considered fixed and replaced by their numerical values, since this effectively reduces the dimension and complexity of the optimization problem that leads to less polyhedral regions in the representation of the PWL solution. This simplified approach can also be taken when it is sufficient to consider only a finite number of values of $(\beta, u_{min}, u_{max})$, since different PWL representations can be pre-computed and stored in real-time computer memory. This facilitates the implementation of reconfiguration mechanisms needed to accommodate actuator failures etc.

IV. Aircraft control allocation examples

We consider three different control configurations of an F18 aircraft

- 5 control surfaces (left and right horizontal tails, left and right ailerons, and rudder) described in [24].

Number of actuators	5	7	10
Number of regions	29	149	637
Number of FLOPS	62	104	118
Stored numbers	499	4907	16007

TABLE I
COMPLEXITY OF THE EXPLICIT PIECEWISE LINEAR SOLUTION FOR CONTROL ALLOCATION PROBLEMS OF DIFFERENT COMPLEXITY.

- 7 control surfaces (left and right horizontal tails, left and right trailing edge flaps, left and right ailerons and rudder), described in [4].
- 10 control surfaces (left and right horizontal tails, left and right trailing edge flaps, left and right aileron, three thrust-vectoring control and rudder), described in [15].

The configuration matrix with 5 control surfaces is

$$\begin{aligned} B_5 &= \begin{pmatrix} 6.47 \cdot 10^{-4} & -6.47 \cdot 10^{-4} & 6.00 \cdot 10^{-4} & -6.00 \cdot 10^{-4} & 5.83 \cdot 10^{-5} \\ -7.35 \cdot 10^{-3} & -7.35 \cdot 10^{-3} & -6.50 \cdot 10^{-4} & -6.50 \cdot 10^{-4} & 0 \\ 0 & 0 & -1.50 \cdot 10^{-4} & 1.50 \cdot 10^{-4} & -6.67 \cdot 10^{-4} \end{pmatrix} \\ u_{min,5} &= (-24, -24, -25, -25, -30)^T \\ u_{max,5} &= (24, 24, 25, 25, 30)^T \end{aligned}$$

and with 7 control surfaces

$$\begin{aligned} B_7 &= \begin{pmatrix} 2.38 \cdot 10^{-4} & -2.38 \cdot 10^{-4} & 1.23 \cdot 10^{-3} & -1.23 \cdot 10^{-3} & 4.18 \cdot 10^{-4} & -4.18 \cdot 10^{-4} & 3.58 \cdot 10^{-5} \\ -6.98 \cdot 10^{-3} & -6.98 \cdot 10^{-3} & 9.94 \cdot 10^{-4} & 9.94 \cdot 10^{-3} & -5.52 \cdot 10^{-4} & -5.52 \cdot 10^{-4} & 0 \\ -3.09 \cdot 10^{-4} & 3.09 \cdot 10^{-4} & 0 & 0 & -1.74 \cdot 10^{-4} & 1.74 \cdot 10^{-4} & -5.62 \cdot 10^{-4} \end{pmatrix} \\ u_{min,7} &= (-24, -24, -8, -8, -30, -30, -30)^T \\ u_{max,7} &= (24, 24, 8, 8, 30, 30, 30)^T \end{aligned}$$

and finally with 10 control surfaces

$$\begin{aligned} B_{10} &= \left(\begin{matrix} -7.64 \cdot 10^{-4} & 7.64 \cdot 10^{-4} & -1.02 \cdot 10^{-3} & 1.02 \cdot 10^{-3} & 2.92 \cdot 10^{-4} \\ -9.30 \cdot 10^{-3} & -9.30 \cdot 10^{-3} & -1.13 \cdot 10^{-3} & -1.13 \cdot 10^{-3} & 0 \\ 1.92 \cdot 10^{-4} & -1.92 \cdot 10^{-4} & 6.82 \cdot 10^{-5} & -6.82 \cdot 10^{-5} & -1.29 \cdot 10^{-3} \end{matrix} \right. \\ &\qquad \left. \begin{matrix} -1.09 \cdot 10^{-3} & 1.09 \cdot 10^{-3} & 5.09 \cdot 10^{-4} & 1.74 \cdot 10^{-7} & 1.74 \cdot 10^{-4} \\ 1.09 \cdot 10^{-3} & 1.09 \cdot 10^{-3} & 1.74 \cdot 10^{-7} & 6.20 \cdot 10^{-3} & 1.74 \cdot 10^{-7} \\ 0 & 0 & 5.23 \cdot 10^{-6} & 1.74 \cdot 10^{-7} & 2.59 \cdot 10^{-3} \end{matrix} \right) \\ u_{min,10} &= (-24, -24, -30, -30, -30, -8, -8, -30, -30, -30)^T \\ u_{max,10} &= (10.5, 10.5, 30, 30, 30, 45, 45, 30, 30, 30)^T \end{aligned}$$

The generalized forces in τ are the three body-axis moments. Minimizing the quadratic criterion $||u||_2^2$ subject to $Bu = \tau$ and $u_{min} \leq u \leq u_{max}$, the mp-QP algorithm gives a PWL function $u^*(\tau)$. This simplified formulation (compared to the more general formulation given in section II) is chosen to allow straightforward comparison with the results in [4], [15], [24]. The complexity of the PWL control allocation functions corresponding to these 3 cases and their computation are summarized in Table I. We compare the real-time computational also with the methods of [24], [4] in Table II and notice that in addition to being optimal (while [4] may be sub-optimal) our approach is typically orders of magnitude more computationally efficient.

Figure 1 shows the underlying partition of the PWL functions $u_i^*(\tau_1, \tau_2, 0)$, $u_i^*(\tau_1, 0, \tau_3)$ and $u_i^*(0, \tau_2, \tau_3)$ for the F18 configuration with 10 control surfaces. Notice that 3 planar intersections of the 3-dimensional domain are shown in order to simplify the illustrations. Each region in the partition corresponds to a different optimal

Number of actuators	5	7	10
Number of real-time FLOPS, mp-QP method	62	104	118
Number of real-time FLOPS, [4]	305	637	1360
Number of real-time FLOPS, [24]	1500	39396	2483040

TABLE II
Comparison of real-time computational complexity of constrained control allocation algorithms.

combination of active constraints (different actuators being saturated in each region). The boundary of the 3D surface in (τ_1, τ_2, τ_3)-space is the feasible set, which equals the attainable moment set as defined in [4], [14]. While the direct allocation method of Durham scales down the optimal control allocation at this boundary along a line to the origin, leading possibly to sub-optimal control allocation, the present approach computes the optimal active constraint combination also when the commanded generalized force is not at the boundary of the feasible set.

V. Marine vessel control allocation - Experimental results

A 1:70 scale model of a dynamically positioned supply ship is considered, Figure 2. Its motion can be controlled by using four independent azimuth thrusters. In this paper the thruster orientations are kept fixed at $\alpha_1 = 43^\circ, \alpha_2 = -43^\circ, \alpha_3 = \alpha_4 = 90^\circ$ such that the control allocation problem is defined in terms of the forces produced by each thruster, $u = (u_1, ..., u_4)$, and the commanded generalized forces in surge, sway and yaw $\tau = (\tau_x, \tau_y, \tau_\psi)$. Hence, $\tau = Bu$, with B defined from (2) as follows, see also [9]:

$$B = \begin{pmatrix} 0.7314 & 0.7314 & 0 & 0 \\ -0.6820 & 0.6820 & 1 & 1 \\ 0.3314 & -0.3314 & 0.3400 & 0.4600 \end{pmatrix} \quad (12)$$

The maximum and minimum forces that can be generated by each thruster is limited

$$u_{max} = -u_{min} = \begin{pmatrix} 0.8 \\ 0.8 \\ 0.1 \\ 0.8 \end{pmatrix} \quad (13)$$

We pose the problem in the form (5)-(8) with the following parameters: $\beta = 0$, $W = I$, and $Q = 10^3 I$. The optimal mp-QP solution is a PWL function represented by 59 polyhedral regions and affine parts. Evaluating the PWL function requires at most 83 arithmetic operations, and storing the search tree and PWL function representation requires 1675 numbers.

In the experiments, the generalized forces are commanded by a generalized DP controller from that was tuned for this model ship [25], [26].

Figures 4 and 5 show experimental results from scenario 1, where the reference position and heading is kept constant. An external environmental force is generated using a wind generator. The force attacks the vessel along the y-axis and is slowly increased throughout the experiment, see Figure 3. We observe that in order to compensate for this disturbance the dynamic positioning controller gradually increases the commanded generalized forces, and after $t \geq 120$ (s) the 3rd thruster is saturated. The constrained control allocation handles this without any inaccuracy in the control allocation, and the tracking performance is consistently excellent throughout the experiment, with positioning errors typically less than 0.01 (m) and heading error less than 2°.

Figures 6 and 7 show experimental results from scenario 2, where the reference heading and position along the x-axis are kept constant at $\psi = 0$, while the reference for the position along the y-axis is changed from 3.5 (m) to 4.5 (m) during the experiment, see Figure 3. The 3rd thruster is saturated at 0.1 (N) during a significant part of the experiment, without leading to loss of performance because the optimal constrained allocation ensures that the commanded generalized force vector τ is still achieved with high accuracy.

VI. Conclusions

We have suggested an optimal approach to constrained control allocation. Comparison with aerospace examples algorithms published in [4], [15], [24], [16], we have demonstrated that the suggested approach gives significant real-time computational advantages, in addition to being optimal. We have also demonstrated the use of the algorithm on ship control examples where we have shown that saturation of one out of four thrusters does not lead to loss of control performance with the constrained control allocation. The general formulation allow several extensions compared to the above mentioned explicit methods, since constraint limits and certain criterion parameters may be taken as parameters to the problem such that the control allocation may be reconfigured in real time simply by changing the point at which the PWL control allocation function is evaluated.

Rate constraints and dynamic optimal constrained control allocation in general can be solved using the same technique. In fact, the explicit model predictive control approach developed in [20], [19] is ideally suited for this purpose. It is worthwhile noticing that the results can be extended to more complicated situations where also the directions of the forces produced by the actuators may be optimized. In this case the problem formulation typically leads to a multi-parametric nonlinear program (mp-NLP) that is harder to solve, but admits approximate solutions with the same implementation advantages as mp-QPs [27].

Appendix A: Multi-parametric quadratic programming

As shown in [20], the mp-QP problem

$$V_z(x) = \min_z \frac{1}{2} z^T H z \tag{14}$$

$$\text{subject to} \quad Gz \leq W + Sx \tag{15}$$

can be solved by applying the Karush-Kuhn-Tucker (KKT) conditions

$$Hz + G^T \lambda = 0, \quad \lambda \in \mathbb{R}^q \tag{16}$$

$$\lambda_i \left(G^i z - W^i - S^i x \right) = 0, \quad i = 1, ..., q \tag{17}$$

$$\lambda \geq 0 \tag{18}$$

$$Gz - W - Sx \leq 0 \tag{19}$$

Superscript i on some matrix denotes the i^{th} row. Assuming H has full rank, (16) gives

$$z = -H^{-1} G^T \lambda \tag{20}$$

Assume for the moment that we know which constraints are active at the optimum for a given x, and let $\tilde{\lambda}$ be the Lagrange multipliers of the active constraints, $\tilde{\lambda} \geq 0$. We can now form matrices $\tilde{G}, \tilde{W}$ and $\tilde{S}$ which contains the rows G^i, W^i and S^i corresponding to the active constraints. Assume that $\tilde{G}$ has full row rank, such that the rows of $\tilde{G}$ are linearly independent. For the active constraints, (17) and (20) gives $-\tilde{G} H^{-1} \tilde{G}^T \tilde{\lambda} - \tilde{W} - \tilde{S}x = 0$, which leads to

$$\tilde{\lambda} = -(\tilde{G} H^{-1} \tilde{G}^T)^{-1} (\tilde{W} + \tilde{S}x). \tag{21}$$

Eq. (21) can now be substituted into (20) to obtain

$$z = H^{-1} \tilde{G}^T (\tilde{G} H^{-1} \tilde{G}^T)^{-1} (\tilde{W} + \tilde{S}x). \tag{22}$$

We have now characterized the solution to (14)-(15) for a given optimal active set, and a fixed x. However, as long as the active set remains optimal in a neighborhood of x, the solution (22) remains optimal, when z is viewed as a function of x. Next, we characterize the region where this active set remains optimal. First, z must remain feasible (19)

$$G H^{-1} \tilde{G}^T (\tilde{G} H^{-1} \tilde{G}^T)^{-1} (\tilde{W} + \tilde{S}x) \leq W + Sx. \tag{23}$$

Second, the Lagrange multipliers λ must remain non-negative (18)

$$-(\tilde{G}H^{-1}\tilde{G}^T)^{-1}(\tilde{W}+\tilde{S}x) \geq 0. \tag{24}$$

The inequalities(23) and (24) describe a polyhedron in the parameter space. This region is denoted as the **critical region** CR_0 corresponding to the given set of active constraints. [20] showed that when you pick an arbitrary $x_0 \in X$ and let (z_0, λ_0) be the corresponding values satisfying the KKT conditions, then one can find the critical region CR_0 from (23) and (24). This region is a convex polyhedral set and represents the largest set of parameters x such that the combination of active constraints at the minimizer remains optimal.

An algorithm has been developed in [18], [19] for constructing polyhedral partitions of the parameter space that explicitly defines the PWL function $z^*(x)$. Below, we give a simplified description of the algorithm, while a more comprehensive description and analysis that also covers degeneracy and infeasibility is found in [19]:

Algorithm 1 (offline mp-QP solver)

1. Initialize the list of unexplored active sets $\mathcal{U}$ with an arbitrary (but feasible) active set. Initialize the list of explored active sets $\mathcal{E}$ to be empty.

2. Choose an arbitrary active set in $\mathcal{U}$, compute the associated linear state feedback (22), Lagrange multiplier (21) and polyhedral region CR_0 defined by (23) and (24). Remove the active set under consideration from $\mathcal{U}$ and add it to $\mathcal{E}$.

3. If $CR_0 = \varnothing$, go to step 2, otherwise go to step 4.

4. For each facet of the corresponding polyhedral representation determine the active set in the neighboring region as described in detail in [19]. For each new active set (i.e. not already in $\mathcal{E} \cup \mathcal{U}$), add it to $\mathcal{U}$.

5. If $\mathcal{U}$ is non-empty, go to 2, otherwise terminate.

Appendix B: Real-time computations using binary search tree

PWL functions defined on polyhedral partitions can be evaluated efficiently by searching a binary search tree constructed using the recently developed method in [22], [23]. Assume the PWL function is $z = z^*(x)$. The key problem is then to determine in which polyhedral region any given x belongs. In order to solve this problem efficiently we design a binary search tree. At each level in the search tree the linear equation corresponding to a hyperplane is evaluated. Based on the sign, one knows on which side of the hyperplane the given x is. If the hyperplane is such that it roughly divides the set of polyhedral regions into two parts with the same number of regions, one has effectively reduced the number of candidate polyhedral regions by half. Only in special cases can one expect to be able to find a hyperplane that splits a given set of polyhedral regions into half, but the method in [22], [23] typically splits the polyhedral regions into 3 approximately equal parts; one on each side of the hyperplane, and one third containing polyhedra that are intersected by the hyperplane. Hence, one can exclude typically about 1/3 of the polyhedral regions by evaluating a single hyperplane at each level in the search tree. The computational complexity of binary tree search is still logarithmic in the number of polyhedral regions [28] and therefore leads to extremely efficient evaluation of such PWL functions.

References

[1] O. J. Sørdalen, "Optimal thrust allocation for marine vessels," *Control Engineering Practice*, vol. 5, pp. 1223–1231, 1997.

[2] T. I. Fossen, *Marine Control Systems: Gudiance, Navigation and Control of Ships, Rigs and Underwater Vehicles*, Marine Cybernetics, Trondheim, 2002.

[3] W. C. Durham, "Constrained control allocation," *J. Guidance, Control and Dynamics*, vol. 16, pp. 717–725, 1993.

[4] W. C. Durham, "Constrained control allocation: Three-moment problem," *J. Guidance, Control and Dynamics*, vol. 17, pp. 330–336, 1994.

[5] W. Jackson, M. P. J. Fromherz, D. K. Biegelsen, J. Reisch, and D. Goldberg, "Constrained optimization based control of real time large-scale systems: Airjet object movement system," in *Proc. IEEE Conf. Decision and Control, Orlando*, 2001.

[6] D. Enns, "Control allocation approaches," in *Proc. AIAA Guidance, Navigation and Control Conference and Exhibit, Boston MA*, 1988, pp. 98–108.

[7] M. Bodson, "Evaluation of optimization methods for control allocation," *J. Guidance, Control and Dynamics*, vol. 25, pp. 703–711, 2002.

[8] K.-P. Lindegaard and T. I. Fossen, "Fuel efficient control allocation for surface vessels with active rudder usage: Experiments with a model ship," *IEEE Trans. Control Systems Technology*, vol. 11, to appear.

[9] S. P. Berge and T. I. Fossen, "Robust control allocation of overactuated ships; experiments with a model ship," in *IFAC Conference on Maneuvering and Control of Marine Craft, Brijuni, Croatia*, 1997.

[10] I. Lindfors, “Thrust allocation methods for the dynamic positioning system,” in *Proc. 10th International Ship Control Symposium, Ottawa, Canada*, 1993, pp. 3.93–3.106.

[11] W. C. Webster and J. Sousa, “Optimum allocation for multiple thrusters,” in *Proceedings International Society of Offshore and Polar Engineers Conference (ISOPE-99), Brest, France*, 1999.

[12] O. Härkegård, “Efficient active set algorithms for solving constrained least squares problems in aircraft control allocation,” in *Proc. IEEE Conf. Decision and Control, Las Vegas NV*, 2002.

[13] T. A. Johansen, T. I. Fossen, and S. P. Berge, “Constrained nonlinear control allocation using quadratic programming,” unpublished, submitted for publication, 2002.

[14] W. C. Durham, “Attainable moments for the constrained control allocation problem,” *J. Guidance, Control and Dynamics*, vol. 17, pp. 1371–1373, 1994.

[15] K. A. Bordignon and W. C. Durham, “Closed-form solutions to constrained control allocation problems,” *J. Guidance, Control and Dynamics*, vol. 18, pp. 1000–1007, 1995.

[16] W. C. Durham, “Efficient, near-optimal control allocation,” *J. Guidance, Control and Dynamics*, vol. 22, pp. 369–372, 1999.

[17] J. A. M. Petersen and M. Bodson, “Fast implementation of direct allocation with extension to coplanar controls,” *J. Guidance, Control and Dynamcis*, vol. 25, pp. 464–473, 2002.

[18] P. Tøndel, T. A. Johansen, and A. Bemporad, “An algorithm for multi-parametric quadratic programming and explicit MPC solutions,” in *Proc. IEEE Conf. Decision and Control, Orlando*, 2001, pp. TuP11–4.

[19] P. Tøndel, T. A. Johansen, and A. Bemporad, “An algorithm for multi-parametric quadratic programming and explicit MPC solutions,” *Automatica*, vol. 39, 2003.

[20] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, “The explicit solution of model predictive control via multiparametric quadratic programming,” in *Proc. American Control Conference, Chicago*, 2000, pp. 872–876.

[21] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, “The explicit linear quadratic regulator for constrained systems,” *Automatica*, vol. 38, pp. 3–20, 2002.

[22] P. Tøndel, T. A. Johansen, and A. Bemporad, “Evaluation of piecewise affine control via binary search tree,” *Automatica*, vol. 39, 2003.

[23] P. Tøndel, T. A. Johansen, and A. Bemporad, “Compuation and approximation of piecewise affine control via binary search tree,” in *IEEE Conf. Decision and Control, Las Vegas, NV*, 2002, pp. ThM12–2.

[24] K. A. Bordingnon and W. C. Durham, “Null-space augmented solutions to constrained control allocation problems,” in *AIAA Guidance, Navigation and Control Conference, Baltimore, MD*, 1995, pp. 328–333.

[25] T. I. Fossen and J. P. Strand, “Passive nonlinear observer design for ships using Lyapunov methods: Experimental results with a supply vessel,” *Automatica*, vol. 35, pp. 3–16, 1999.

[26] J. P. Strand, *Nonlinear Position Control Systems Design for Marine Vessels*, Ph.D. thesis, Norwegain University of Science and Technology, Trondheim, 1999, http://www.itk.ntnu.no/ansatte/Fossen_Thor/PhD/strand.pdf.

[27] T. A. Johansen, “On multi-parametric nonlinear programming and explicit nonlinear model predictive control,” in *IEEE Conf. Decision and Control, Las Vegas, NV*, 2002, pp. ThA12–5.

[28] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Data structures and algorithms*, Addison Wesley, 1983.

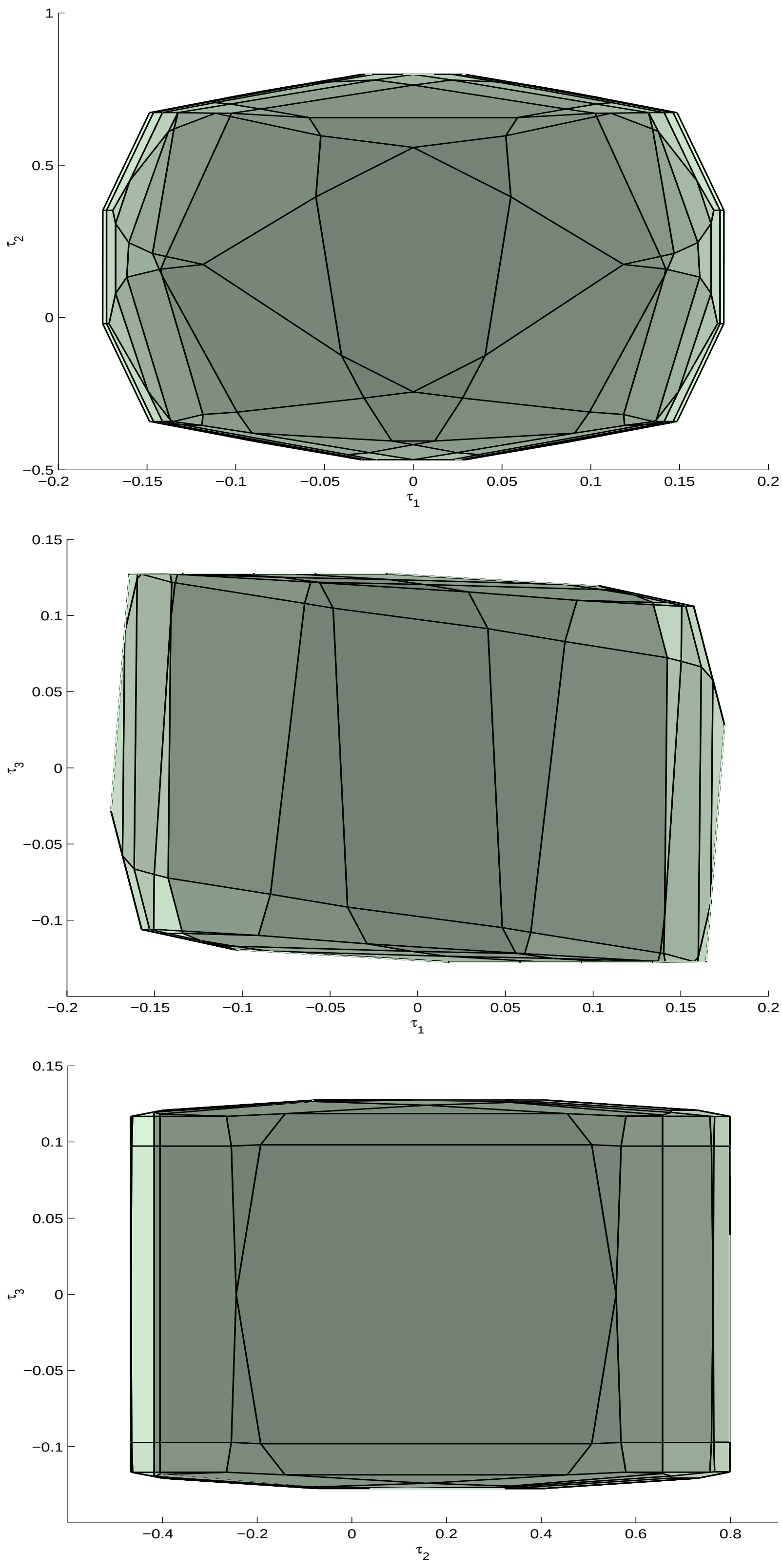

Fig. 1. Planar intersection $(\tau_1, \tau_2, 0)$, $(\tau_1, 0, \tau_3)$ and $(0, \tau_2, \tau_3)$ of the underlying partition of the PWL optimal control allocation for the F18 configuration with 10 control surfaces.

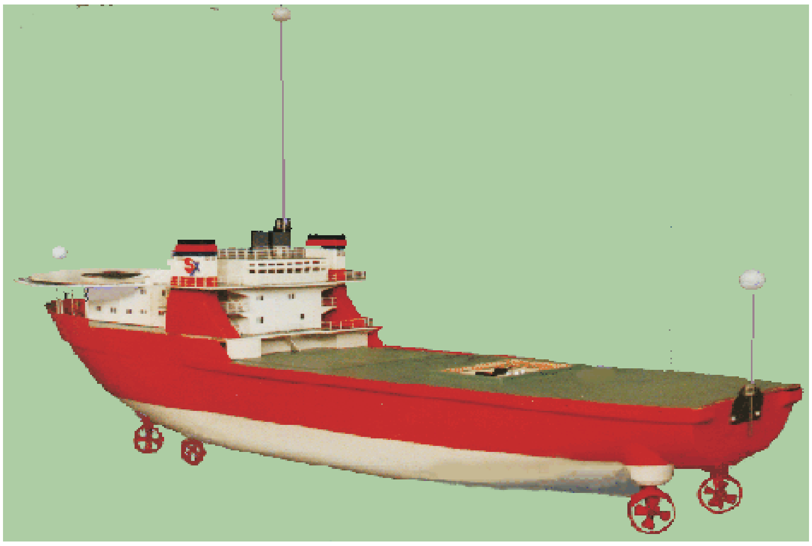

Fig. 2. Picture of Cybership I, a 1:70 scale model of a supply ship with 4 azimuth thrusters.

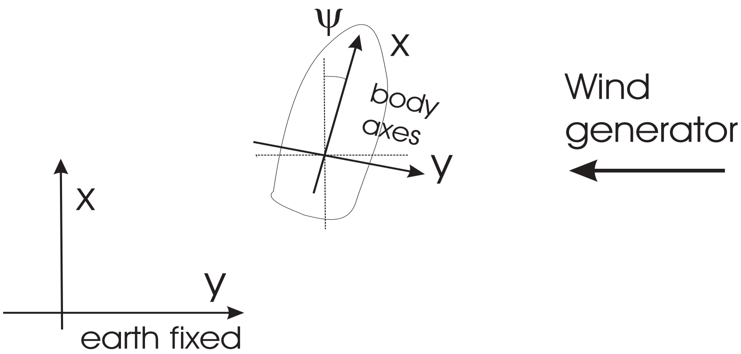

Fig. 3. Definition of coordinate systems.

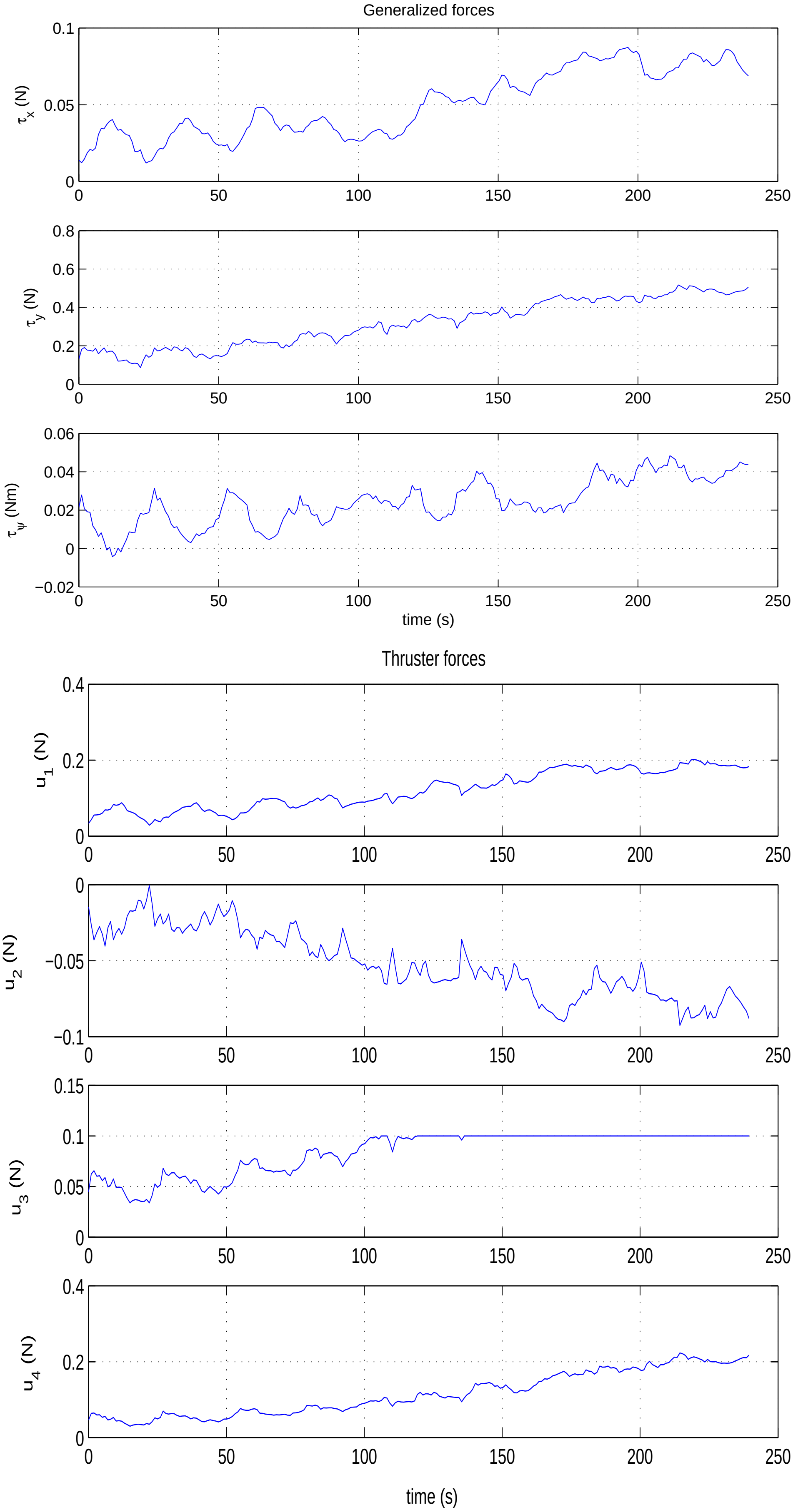

Fig. 4. Experimental results, scenario 1. The curves show generalized forces, and thruster forces computed using the optimal constrained thrust allocation. Notice that u_3 saturates which u_1, u_2 and u_4 compensates this.

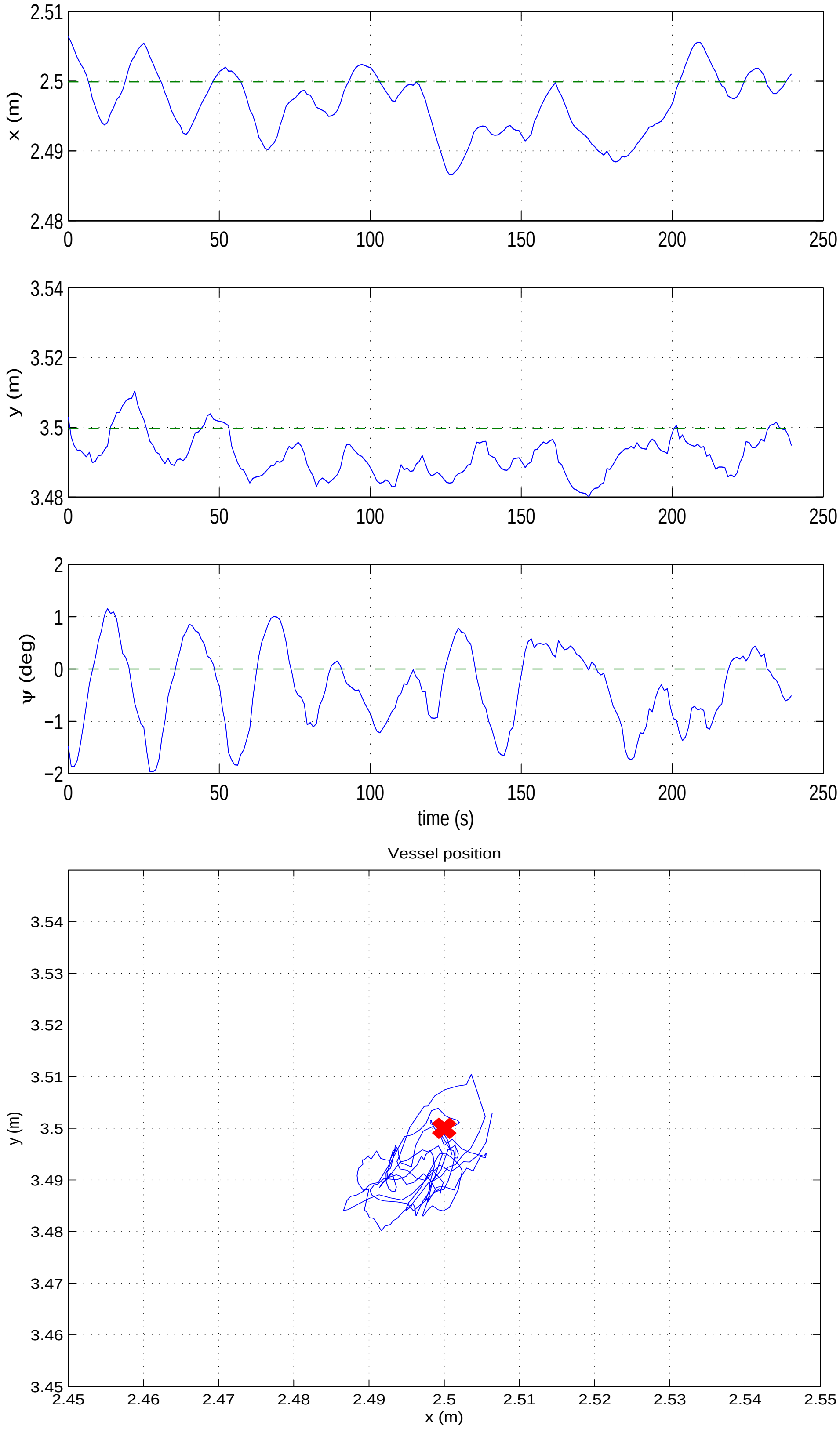

Fig. 5. Experimental results, scenario 1. The curves show position and heading, with dashed curves representing the reference. Lower plot shows dynamic positioning station-keeping performance.

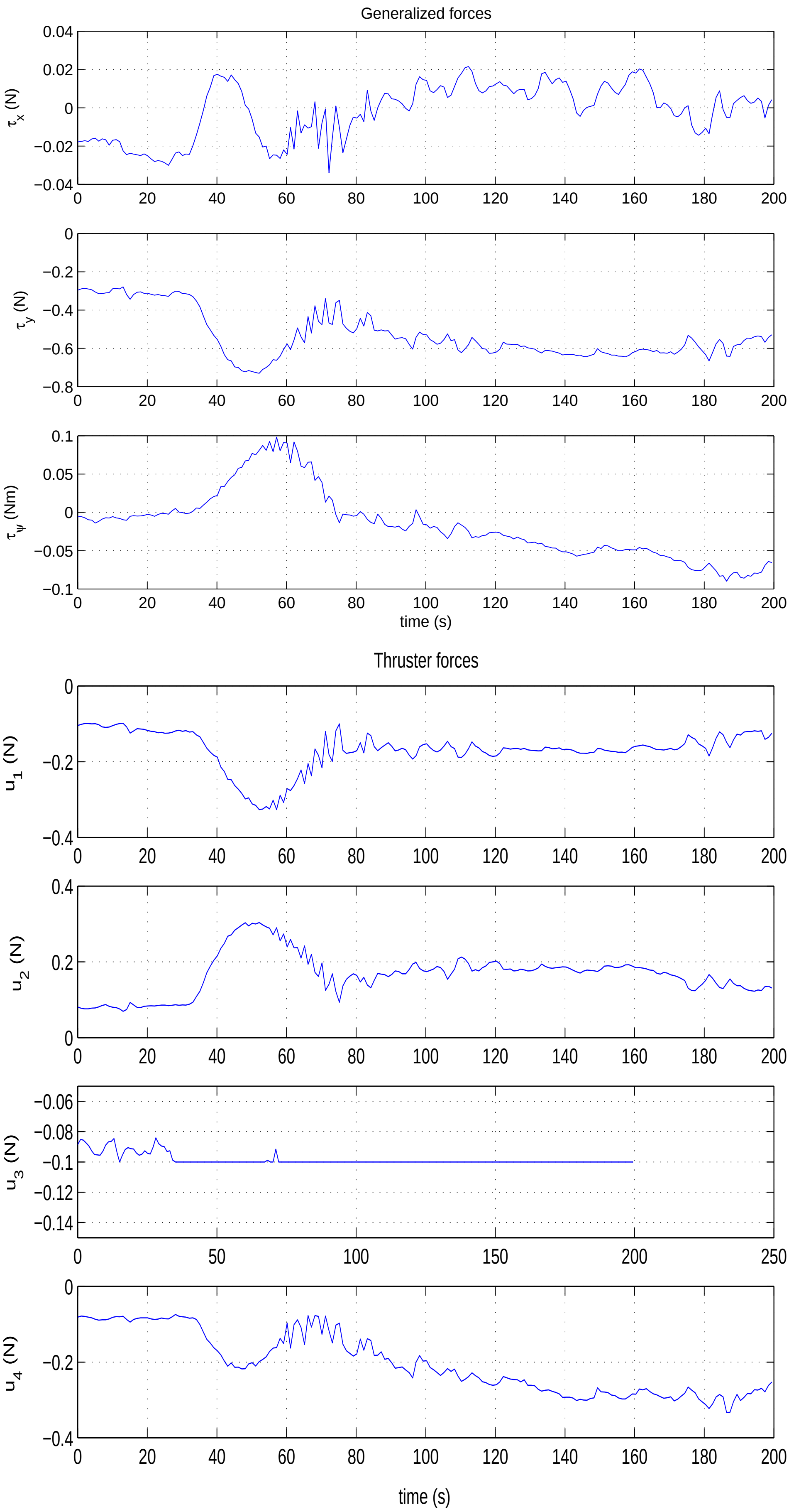

Fig. 6. Experimental results, scenario 2. The curves show generalized forces, and thruster forces computed using the optimal constrained thrust allocation. Notice that u_3 saturates which u_1, u_2 and u_4 compensates this.

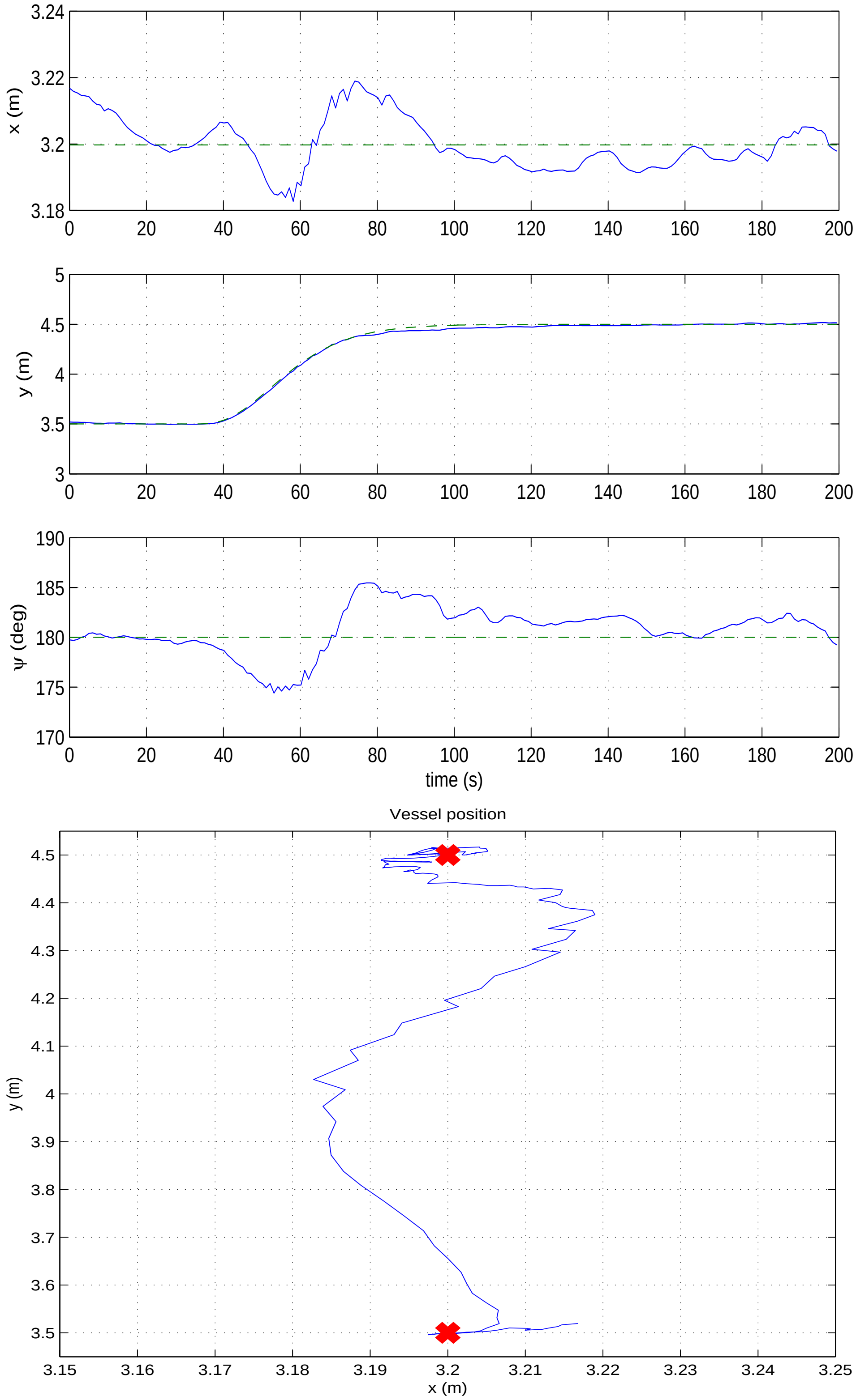

Fig. 7. Experimental results, scenario 2. The curves show position and heading, with dashed curves representing the reference. Lower plot shows dynamic positioning station-keeping performance.