Fault Tolerant Control Allocation for a Thruster-Controlled Floating Platform using Parametric Programming

Jørgen Spjøtvold* and Tor A. Johansen*

***Abstract*— The task in control allocation is to determine how to generate a specified generalized force from a redundant set of control effectors where the associated actuator control inputs are constrained, and other physical and operational constraints and objective should be met. In this paper we consider a convex approximation to a control allocation problem for a *thruster-controlled floating platform*. The platform has eight rotatable azimuth thrusters and the high level controller is assumed to specify three generalized forces; surge, sway and yaw. The control allocation problem is formulated as a convex quadratic- or linear program, where the constraints are dependent on the specified generalized force. The problem is solved explicitly by viewing the generalized forces as a vector of parameters and utilizing parametric programming techniques. For convex parametric quadratic programs (pQP) or parametric linear programs (pLP) with a linear parametrization of the constraints, there always exists a continuous piecewise affine (PWA) minimizer function. Consequently, the conventional on-line optimization can be replaced by a simple evaluation of a PWA function. Experimental results for a scale model of a platform are presented. It is shown how thruster failure scenarios can be handled by automatic reconfiguration of the control allocation, exploiting symmetry of the thruster configuration.**

***Index Terms*— Parametric Quadratic Programs. Optimal Thrust Allocation. Marine Control.**

I. Introduction

The task in control allocation is to determine how to generate a specified generalized force from a redundant set of actuators, motors and control effectors where the associated controls are constrained, see e.g. [1]–[11]. Control effectors are the force producing surfaces and devices, such as rudders and propellers, while actuators are the devices that set the position or orientation of the effectors. The main objective is to obtain the desired generalized force, however, it is also common to incorporate secondary objectives, such as minimizing power consumption, power transients and mechanical tear and wear. Several other factors, such as actuator- and control effector-dynamics, and physical and operational constraints, should also be incorporated. One way of achieving these secondary goals is to solve a constrained optimization problem online at every sampling instant. A control allocation approach to control synthesis often has the advantage that a high level control law that is independent of actuator and effector configuration can be designed. The approach also utilizes redundancy of the effectors to obtain fault tolerant control.

Only recently, it has been proposed to solve the constrained optimization problem off-line [6], [7] by utilizing parametric programming techniques [12]–[14]. For certain classes of allocation problems the online computational effort then reduces to an evaluation of a piecewise affine function. The main advantages of this approach are: $i)$ removing the need for sophisticated optimization software on the processor, $ii)$ the correctness of the solution can be verified off-line, which is a key issue in safety critical applications, $iii)$ the worst case number of arithmetic operations needed to find the solution can easily be computed, $iv)$ the average and worst case number of arithmetic operations needed to find the solution is usually greatly reduced, and $v)$ evaluation of the PWA function can be implemented using fixed point arithmetic. The main drawbacks, on the other hand, are that $i)$ the problem class for which this solution strategy is applicable is limited, and in cases where an exact solution can be found $ii)$ obtaining an explicit solution may require excessive off-line computations, and $iii)$ the storage space required to represent the solution may exceed the available memory. The drawbacks may in particular be apparent when the system has to accommodate reconfiguration due failure situations or operation in several modes. However, if we are able to obtain and represent an explicit solution, it is clearly more desirable than utilizing online optimization.

In this paper we focus on optimal thrust allocation for a scale model of a thruster-controlled floating platform that is commonly used for offshore oil drilling, production, accommondation and heavy-lift operations. In particular, we seek to obtain an explicit solution to the control allocation problem. The platform has eight rotatable fixed pitch azimuth thrusters and the high level controller specifies surge, sway and yaw forces. The task is to determine the thrust magnitude and azimuth angle for each thruster such that the desired surge, sway and yaw forces are generated. Each thruster can physically rotate 360 degrees, but the thrust magnitude is limited. In addition, it is necessary to enforce operational constraints on the azimuth angles to avoid interaction between the thrusters leading to thrust losses.

Current non-optimization based approaches to this problem are either conservative in terms of utilizing only a limited fraction of the attainable force set[1] or not being optimal

* Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway. e-mail: torj@itk.ntnu.no. Jørgen Spjøtvold is currently with ABB.

This research is part of the Strategic University Program on Computational Methods for Nonlinear Motion Control funded by the Research Council of Norway.

[1]The attainable force set is the set of generalized forces that can be generated by the control effectors (azimuth thrusters) while fulfilling the constraints.

in terms of power consumption. One approach is to fix the azimuth thrusters' angles and use a pseudo inverse to compute the thrust magnitudes [15]; a method that only utilizes a limited volume of the attainable force set and may yield singular thruster configurations if thruster failure occurs. Another approach is to use a generalized inverse to compute both azimuth angles and thrust magnitudes [11]. This approach is also conservative with respect to the attainable force set and not optimal with regards to power consumption. Neither of the approaches efficiently takes into account the constraints on the control inputs other than saturating the controls, and consequently an optimization based approach should be beneficial both with regards to minimizing power consumption and utilizing a larger volume of the attainable force set.

Optimization based approaches, both explicit solutions and online optimization, have been successfully tested on ships and other control allocation problems [1]–[11]. However, the platform has 8 control effectors and 16 controls inputs and needs to handle thruster failure conditions, making it questionable whether obtaining an explicit solution is computationally tractable. A further concern of online optimization based on numerical methods is its reliability, due to the safety-critical continuous operation of such platforms. We seek to demonstrate that obtaining an explicit solution to an approximation of the control allocation problem for the platform is computationally tractable, that the explicit solution can be verified and evaluated at a high frequency, and that the performance is satisfactory even if rate constraints for the effectors are neglected.

II. Static Constrained Control Allocation.

In this paper we consider discrete-time systems on the form:

$$x^+ = f(x, \bar{\tau}, t), \tag{1a}$$
$$\bar{\tau} = g(x, u, t), \tag{1b}$$

where $x \in \mathbb{R}^{n_x}$ is the state, $x^+ \in \mathbb{R}^{n_x}$ is the successor state, $\bar{\tau} \in \mathbb{R}^{n_\tau}$ is the generalized force, t is time and $u \in \mathbb{R}^{n_u}$ is the control. Assume further that there exists a virtual time-varying feedback controller

$$\tau := k(t, x), \tag{2}$$

that is, our desired generalized force is τ while the actual generalized force is $\bar{\tau}$. The task in control allocation is to determine controls u, satisfying some constraints $u \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$, that generate the generalized force $\bar{\tau}$ that is in some sense closest to the desired τ. When it is possible to obtain $\bar{\tau} = \tau$, it is often an uncountable number of combinations of controls that achieve the desired generalized force. Hence, in this case, secondary objectives such as minimization of power consumption and actuator tear and wear are considered.

It is common to assume a linear relationship between the controls and the generalized forces [1]–[4], [6]

$$\bar{\tau} = Bu.$$

In this formulation it is still possible to accommodate for certain actuator and control effector nonlinearities through a nonlinear mapping, as will be done in our application through a quadratic relation between the thruster force and propeller speed.

The simplest version of the constrained control allocation problem is to find a solution (u, s) to the equations

$$\tau + s = Bu \quad \text{and} \quad u \in \mathcal{U}, \tag{3}$$

such that the slack variable $s = 0$ if there exists $u \in \mathcal{U}$ for which $\tau = \bar{\tau} = Bu$.

There exists several methods for the purpose of solving this problem and the reader is referred to [1] and references therein for details. We will use the following formulation:

$$\mathbb{P}(\tau): \; J^*(\tau) := \inf_{u,s} \{ J(u, s, \tau) \mid (u, s, \tau) \in \mathcal{Z} \}, \tag{4a}$$
$$J(u, s, \tau) := \|Qs\|_l + \|Ru\|_l, \tag{4b}$$
$$\mathcal{Z} := \left\{ (u, s, \tau) \;\middle|\; \begin{array}{l} Bu + s = \tau, \\ u \in \mathcal{U} \end{array} \right\}, \tag{4c}$$

where $(u, s, \tau) \in \mathbb{R}^{n_u} \times \mathbb{R}^{n_\tau} \times \mathbb{R}^{n_\tau}$, $Q \geq 0$ and $R \geq 0$ are weight matrices, respectively penalizing use of controls and deviation from desired generalized force, $\mathcal{U}$ is the constraint set on the control inputs, and $l \in \{1, 2, \infty\}$[2] denotes the weighting norm.

Remark 1: Please note that it is possible to obtain an exact penalty function [16] (in the sense that if there exists $u^* \in \mathcal{U}$ such that $Bu^* = \tau$, then $s^* = 0$) by utilizing mixed norms in the objective function, e.g. $J(u, s, \tau) = \|Qs\|_1 + \|Ru\|_2$. This, however often yields a convex (as opposed to strictly convex) problem, which is slightly more complicated to solve explicitly [17]–[19] and provides little practical benefit.

III. Explicit Solutions to Linear Constrained Control Allocation via Parametric Programming.

A. Problem Setup

Under certain assumptions on the control allocation problem (4), recent progress in parametric programming allows it to be solved explicitly, yielding a piecewise affine solution function [6]. In the parametric programming setup we have that $\mathbb{P}(\tau)$ is to be solved for all values of $\tau \in \mathcal{T}$, where $\mathcal{T}$ is the domain of $J^*(\cdot)$.

Define the set-valued maps $\mathcal{Y} : \mathcal{T} \to 2^{\mathbb{R}^{n_u} \times \mathbb{R}^{n_\tau}}$

$$\mathcal{Y}(\tau) := \{ (u, s) \mid (u, s, \tau) \in \mathcal{Z} \}$$

and $\mathcal{Y}^* : \mathcal{T} \to 2^{\mathbb{R}^{n_u} \times \mathbb{R}^{n_\tau}}$

$$\mathcal{Y}^*(\tau) := \arg\min_{(u,s)} \{ J(u, s, \tau) \mid (u, s, \tau) \in \mathcal{Z} \}.$$

In the sequel we let $y := [u^T \; s^T]^T$ and let $y^*(\cdot)$ denote a selection of $\mathcal{Y}^*(\cdot)$, that is, $y^*(\tau) \in \mathcal{Y}^*(\tau)$ for all $\tau \in \mathcal{T}$.

[2] $l = 2$ denotes, with some abuse of mathematical rigor, the quadratic norm, that is, $\|Qx\|_2 := x^T Q x$.

B. Solution via Parametric Programming

One might distinguish between two types of linear allocation problems; $i)$ where the set $\mathcal{U}$ is convex, and $ii)$ when $\mathcal{U}$ is non-convex. However, in this paper, we consider only the case where $\mathcal{U}$ is convex, closed and polyhedral. This immediately implies that (4) is a convex problem and that it attains its minimum for all $\tau \in \mathcal{T}$. By using linear or quadratic norms in the cost function it is evident that $\mathbb{P}(\cdot)$ is either a parametric linear program (pLP) or parametric quadratic program (pQP). These problems have been subject to a vast amount of research, e.g. [13], [14]. In parametric programming the goal is to divide the set of parameters of interest into a set of smaller regions such that each region is associated with a function that is optimal for the optimization problem when restricted to its region. For convenience we summarize the solution properties specialized to our problem formulation:

Theorem 1 (pLPs and pQPs [13]): Consider problem (4) and let $\mathcal{U}$ be a closed polyhedron.

(i) If $l \in \{1, 2, \infty\}$, then $J^*(\cdot)$ is continuous and convex.
(ii) If $l \in \{1, 2, \infty\}$, then there exists a continuous selection $y^*(\cdot)$ of $\mathcal{Y}^*(\cdot)$ that is piecewise affine. Moreover, if $R > 0, Q > 0$ and $l = 2$, then $y^*(\cdot)$ is unique.
(iii) If $l \in \{1, \infty\}$, then $J^*(\cdot)$ is piecewise affine.
(iv) If $l = 2$, then $J^*(\cdot)$ is piecewise quadratic.

Both for pLPs and pQPs the optimizer is PWA and consequently the most common approach for solving the allocation problem where online-optimization is utilized can be substituted with an evaluation of a PWA function [6].

C. Reconfigurable control allocation.

In many applications it is desirable to be able to switch on and off effectors or to change the constraints imposed on the control inputs to an effector. Reasons for this might be handling of actuator/effector failure and different operational modes. The most straightforward way of achieving this is to define additional parameters ϕ, and rewrite (4) as

$$J^*(\tau, \phi) := \min_{(u,s)} \left\{ \|Qs\|_l + \|Ru\|_l \mid (u, s, \tau, \phi) \in \mathcal{Z}_\phi \right\},$$
$$\mathcal{Z}_\phi := \{(u, s, \tau, \phi) \mid Bu + s = \tau, u \in \mathcal{U}(\phi)\}.$$

This approach does not significantly complicate the online optimization problem. In addition, if the parametrization $\mathcal{U}(\cdot)$ is linear, it is possible to solve the problem explicitly [6]. However, with parametric programming the complexity of the optimal control $u^*(\cdot, \cdot)$ may become too high for the available memory as solution complexity scales quickly in the number of parameters.

IV. Case study

In this section we present the problem formulation and experimental results for static control allocation for a scale model of a thruster-controlled floating platform, see Figure 1. The high level controller sending commands to the thrust allocation may be dynamic positioning, joystick control or thruster assisted position mooring control [15].

Fig. 1. CyberRig I: Scale model of a thruster controlled platform with 8 azimuthing fixed pitch thrusters.

We first illustrate how to obtain an explicit solution to the control allocation problem when the high level controller specifies surge, sway and yaw forces and the optimization problem is convex. Secondly, we show how a selected set of thruster or power failure situations can be handled. National and international regulations [20] require that the control system is operable after any single point failure, such as loss of a single thruster, single diesel generator or electric switchboard.

A. System Description

The control system takes into account surge-, sway-, and yaw-motions, with the corresponding vessel fixed generalized forces $\tau := [X, Y, N]^T$. Assume that the vessel has a set $\mathcal{P} := \{p_1, p_2, \ldots, p_I\}$ of rotatable thrusters such that each device has two controls; direction and thrust magnitude. The thruster indexed by i is located at the position $r_i := [l_{i,x}\ l_{i,y}\ l_{i,z}]^T$ relative to the center of rotation in the vessel fixed coordinate system. Assume further that the force T_i from the i^{th} thruster is limited to the x-y plane in the vessel fixed coordinate system. Thruster p_i then produces a force T_i in the direction defined by the angle α_i. The contribution of the i^{th} thruster to the generalized forces acting on the vessel is given by:

$$X_i := T_i \cos \alpha_i, \tag{5a}$$
$$Y_i := T_i \sin \alpha_i, \tag{5b}$$
$$N_i := T_i \left(l_{i,x} \sin \alpha_i - l_{i,y} \cos \alpha_i \right). \tag{5c}$$

In addition we have that each azimuth angle α_i and thrust force T_i are constrained to the sets

$$\mathcal{O}_i := \left\{ (\alpha_i, T_i) \,\middle|\, \underline{\alpha}_i \le \alpha_i \le \overline{\alpha}_i, \underline{T}_i \le T_i \le \overline{T}_i \right\}, \quad i = \{1, 2, \ldots, I\},$$

where $\underline{\alpha}_i$, $\overline{\alpha}_i$, $\underline{T}_i$ and $\overline{T}_i$ are lower and upper bounds on the azimuth angle and thrust force for the i^{th} thruster, respectively. We introduce the concept of *attainable force set*[3]:

[3]In the aviation literature the attainable force set is referred to as the attainable moment set [3]

Definition 1 (Attainable Force Set (AFS)): The *attainable force set* for a set of control effectors $\mathcal{P} := \{p_1, p_2, \dots, p_I\}$ is given by

$$\mathcal{T} := \{\tau \in \mathbb{R}^{n_\tau} \mid (\alpha_i, T_i) \in \mathcal{O}_i,\ i \in \{1, 2, \dots, I\}\}.$$

In other words, the AFS is the set of generalized forces that can be generated by the thrusters while fulfilling the constraints. For dynamically positioned offshore vessels the AFS is usually presented as a capability plot illustrating the wind and sea loads the thruster system is able to counteract [21].

The relationship (5) can be written as the non-linear equation

$$\tau = A(\alpha)T,$$

where $\alpha := [\alpha_1, \dots, \alpha_I]^T$ and $T := [T_1, \dots, T_I]^T$. However, to obtain a linear relationship we follow the procedure in [11] where the concept of extended thrust is introduced. The extended thrust vector is found by decomposing the individual thrust vectors in the horizontal plane according to: $u_{i,x} := X_i, u_{i,y} := Y_i$ and $u_i := [u_{i,x}\ u_{i,y}]^T \in \mathbb{R}^2$. The generalized thrust vector is then given by the linear equation

$$\tau = Bu,$$

where $u := [u_{1,x}\ u_{1,y}\ u_{2,x}\ u_{2,y}\ \dots\ u_{I,x}\ u_{I,y}]^T$ and the matrix B is given by

$$B := \begin{bmatrix} 1 & 0 & \dots & 1 & 0 \\ 0 & 1 & \dots & 0 & 1 \\ -l_{1,y} & l_{1,x} & \dots & -l_{I,y} & l_{I,x} \end{bmatrix}.$$

It is easy to see that the sets $\{\mathcal{O}_i\}_{i=1}^I$ translate into constraints on the controls $\{u_i\}_{i=1}^I$ defined by

$$u_i \in \mathcal{C}_i := \left\{ [u_{i,x}\ u_{i,y}]^T \;\middle|\; \begin{array}{l} u_{i,x} = T_i \cos\alpha_i, \\ u_{i,y} = T_i \sin\alpha_i, \\ (\alpha_i, T_i) \in \mathcal{O}_i \end{array} \right\}, \quad i = \{1, 2, \dots, I\}.$$

We will refer to the sets $\{\mathcal{C}_i\}_{i=1}^I$ as attainable thrust regions; the set of surge and sway forces that can be generated by a single thruster.

Definition 2 (Attainable Thrust Region (ATR)): The *attainable thrust region* for a set of I thrusters is given by[4] $\mathcal{C} := \bigoplus_{i=1}^I \mathcal{C}_i$.

Hence, the ATR is the set of surge and sway forces that can be generated by a set of thrusters.

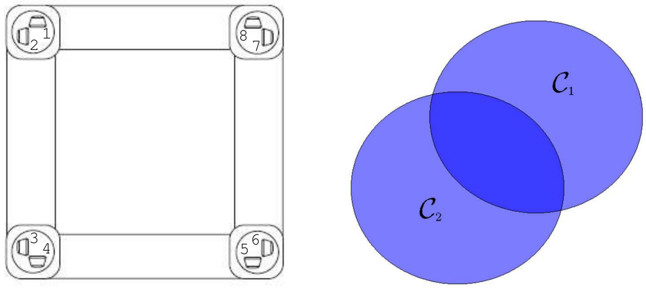
1) Thruster Model: In this paper we utilize a conventional quadratic thruster characteristic [15], that is, the thrust force from a given thruster is given by

$$T = K_T \rho D^4 |n| n =: \gamma(K_T, \rho, n) \tag{6}$$

where K_T is a strictly positive thrust coefficient, ρ is the water density, D is the propeller diameter, and n is the propeller speed. Assuming constant water density and thrust coefficient we see that $\gamma(\cdot)$ is reduced to a function only of the propeller speed. Consequently, if for the i^{th} thruster the desired extended thrust vectors are X_i and Y_i, we recover the thrust force T_i and azimuth angle α_i from the relationships (5) and the propeller speed from (6). Note that it is straightforward to replace (6) by a more advanced thruster characteristic.

The flow chart for constrained control allocation for the floating platform can be represented as depicted in Figure 2. Hence our task is to compute some optimal $u^*(\tau)$ when τ is given.

2) Thruster configuration: The thruster configuration for the floating platform is depicted in Figure 3(a). Since the thrusters are speed controlled, we have $\underline{T}_i = 0$ leading to convex AFS.



(a) Thruster configuration for CyberRig I. Thruster 1 and 2 are situated on leg 1, thruster 3 and 4 on leg 2 etc.

(b) Translated ATRs for the two thrusters on leg 1.

Fig. 3. Thruster configuration and translated ATRs.

Two azimuth thrusters are placed on each of the four legs and each thruster can rotate 360 degrees. It is not straightforward to obtain the AFS for the vessel due to the following: Considering one leg of the platform, the two thrusters are positioned such that one thruster may affect the flow pattern around the other thruster, resulting in loss of thrust and non-linear behavior such that K_T in (6) depends on the azimuth and speed of the neighbouting thruster. In Figure 3(b) we have illustrated this interaction by translating the ATRs for the two thrusters on leg one to their physical location on the vessel.

To avoid this interaction, sectors are introduced that are mutually exclusive in the sense that if thruster p_i produces a force in direction $\alpha_i \in \mathcal{S}_i$, then p_{i+1} is not allowed to produce a force in direction $\alpha_{i+1} \in \mathcal{S}_{i+1}$, where $i \in \{1, 3, 5, 7\}$. See Figures 3(b) and 4(a) for an illustration. More precisely,

$$\alpha_i \in \mathcal{S}_i \Rightarrow \alpha_{i+1} \notin \mathcal{S}_{i+1}, \quad i \in \{1, 3, 5, 7\}.$$

The most straightforward approach that may be utilized to meet this operational constraint is to introduce forbidden sectors such that thruster p_i (p_{i+1}) never produce a force in direction $\alpha_i \in \mathcal{S}_i$ ($\alpha_{i+1} \in \mathcal{S}_{i+1}$), where $i \in \{1, 3, 5, 7\}$. This means that p_i can only produce a force in a restricted ATR, that is, $u_i \in \bar{\mathcal{C}}_i := \mathcal{C}_i \backslash \mathcal{S}_i$.

[4]Given two sets $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}^n$, the *Minkowski set addition* is defined as $X \oplus Y := \{x + y \mid x \in X, y \in Y\}$. Given the a sequence of sets $\{X_i\}_{i=a}^b$, we define $\bigoplus_{i=a}^b X_i := X_a \oplus \dots \oplus X_b$.

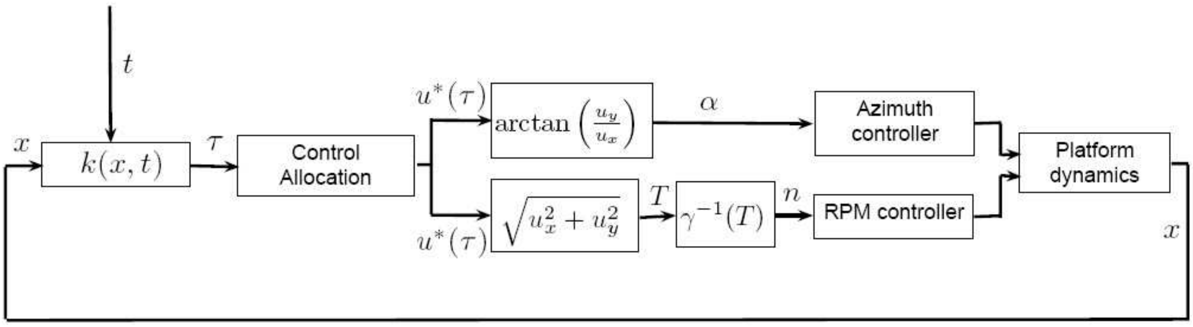
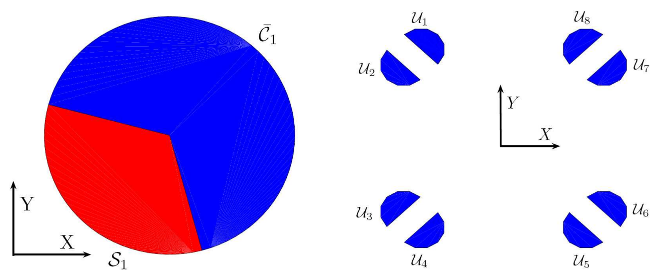


Fig. 2. Flowchart for control and allocation for CyberRig I.



(a) Restricted ATR $\bar{\mathcal{C}}_1$ for top thruster on the top left leg after forbidden sector has been artificially removed. The sector $\mathcal{S}_1$ is removed to avoid unwanted nonlinear interaction with the second thruster on the leg.

(b) Translated and restricted ATRs for all thrusters after forbidden sectors have been removed.

Fig. 4. Approximation of the ATRs.

B. Solution approach

We convexify and approximate the problem by restricting the ATR for each thruster to be an inner polyhedral approximation of a half circle, see Figure 4(b). In the sequel, we let inner approximations of the restricted ATRs $\{\bar{\mathcal{C}}_i\}_{i=1}^I$ be denoted by $\{\mathcal{U}_i\}_{i=1}^I$. We then minimize the thrust magnitude for each thruster and the problem becomes:

$$J^*(\tau) := \min_{u,s} \left\{ \frac{1}{2}\left(u^T R u + s^T Q s\right) \,\middle|\, (u,s,\tau) \in \mathcal{Z} \right\}, \tag{7a}$$

$$\mathcal{Z} := \left\{ (u,s,\tau) \,\middle|\, \begin{array}{l} Bu = \tau + s, \\ u_i \in \mathcal{U}_i,\ i \in \{1,2,\dots,8\} \end{array} \right\}, \tag{7b}$$

where $R = I$, $Q = 10^3 \times I$, and $(u,s,\tau) \in \mathbb{R}^{16} \times \mathbb{R}^3 \times \mathbb{R}^3$. This is clearly a convex optimization problem that can be solved by a single pQP.

C. Experimental Results

1) Nominal operation: The explicit solution to (7) consists of 12522 polyhedral regions. A binary search tree [22] was then constructed for the purpose of evaluating the PWA function. The worst case depth of the search tree was 24, worst case number of arithmetic operations needed to find the solution was 264, and the tree was stored using 4.218.546 numbers, 468.923 being integers and 3.749.623 being real numbers. On a Dell LATTITUDE laptop with a 1.7 GHz Intel Pentium M CPU running Windows XP and MATLAB 7.0 the PWA function could be evaluated at approximately 200 kHz. The QNX real-time system on the experimental scale model platform has a sample frequency of 10 Hz, so the processor was free for other tasks. In Figure 5 we have depicted commanded and measured[5] generalized forces for the case where all desired generalized forces was contained in the feasible part of the AFS. In Figures 7(a) and 8(a) we have depicted the measured azimuth angles and RPMs, respectively. In Figure 6 we let the desired generalized forces be infeasible (that is, either not contained in the AFS or in an infeasible part) in large parts of the time-series. We cannot hope to achieve the desired generalized force in this case, and as expected the constraints are fulfilled, but that the desired generalized force is not obtained. It is simple to prioritize which component (surge, sway or yaw) that is the most important by setting the weights on the slack variables s. In Figures 7(b) and 8(b) we have depicted the measured azimuth angles and RPMs, respectively.

2) Fault tolerant control allocation: As described in Section III-C, fault tolerant control allocation may be computationally demanding if an explicit solution to the problem is desired. However, for this particular application, the symmetry of the problem can be exploited to obtain great reduction in both storage space and required off-line computation.

Consider the case where thruster 1 fails, abbreviated Pf1. The solution to this problem is obtained simply by removing the associated controls u_1 and corresponding constraints from the optimization problem. The question becomes whether solving this problem also give us the solutions to the scenarios where thruster 3, 5, or 7 fail. We argue that this is the case for thruster 3, abbreviated Pf3 (the arguments, with obvious modifications, hold for the other situations as well). Pf3 would be identical to Pf1 if the vessel fixed coordinate was rotated 90 degrees, however, surge and sway forces for each thruster would be defined relative to the rotated coordinate system. Consequently, we can obtain the solution to Pf3 by the following procedure:

(i) Rotate the surge and sway components of τ by 90

[5]Only the RPMs and azimuth angles are measured and the *measured generalized forces* are derived from the inverse relationship in Figure 2.

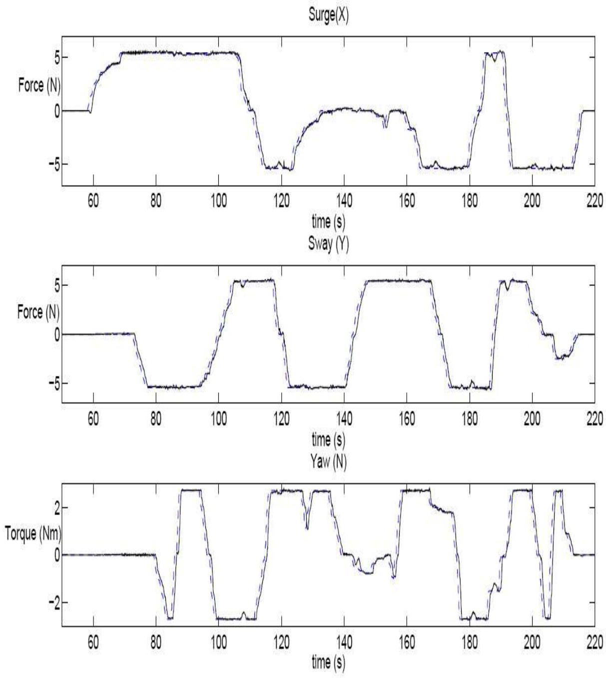


Fig. 5. Experimental results for CyberRig I; control allocation over convex ATRs and desired generalized force was always contained in the AFS. The dotted line is the commanded generalized force and the solid line is measured.

degrees, i.e

$$\tau^r = \begin{bmatrix} R(90)\begin{bmatrix} X \\ Y \end{bmatrix} \\ N \end{bmatrix}.$$

(ii) Evaluate the solution to Pf1 at τ^r, and denoted the solution $u^r(\tau^r)$.

(iii) Rotate each $u_i^r(\tau^r)$, $i = 2, \ldots, 8$, by -90 degrees to obtain $\bar{u}_i^*(\tau^r)$, $i = 2, \ldots, 8$. Let $\bar{u}_i^*$ be the control input to thruster $i+2$, $i = 2, \ldots, 6$ and $\bar{u}_7^*$ and $\bar{u}_8^*$ the control inputs to thrusters 1 and 2, respectively.

Note that this procedure only works because the yaw component (N) of the generalized force is not affected by the rotation of the coordinate system due to the symmetry of the problem when the center of rotation is at the center of the platform.

In Table I we have listed cases for which the solution can be found simply by rotation of another solution. Hence, by considering 3 different failure configurations, we obtain solutions for 12 cases. In Figure 9 we have depicted results for when thruster failure occurs. Assuming the electric power buses are split into four segments, each corresponding to a machine room and switchboard feeding two thrusters in each leg, this covers all critical single point failures; from a single thruster to a whole machine room.

V. Conclusion

A convex constrained control allocation problem was formulated for the purpose of mapping desired generalized forces to control inputs. The explicit PWA solution function

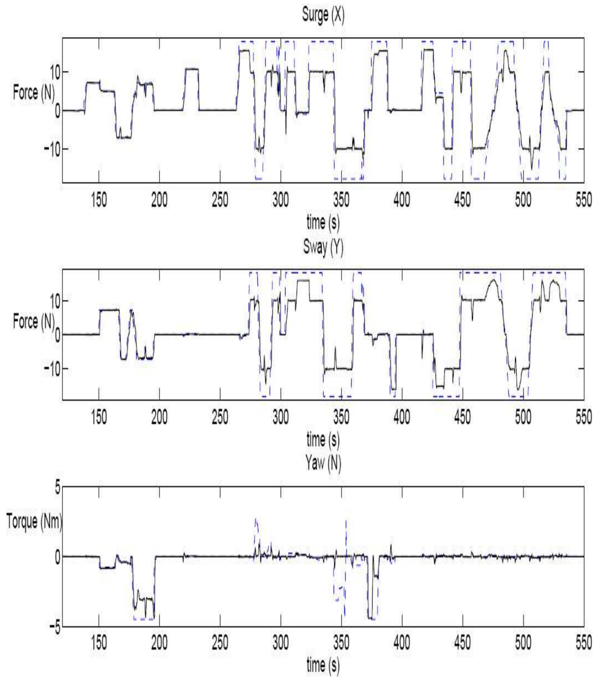


Fig. 6. Experimental results for CyberRig I; control allocation over convex ATRs and desired generalized force was not always contained in the AFS. The dotted line is the commanded generalized force and the solid line is measured.

Thruster failure	Rotated 90	Rotated 180	Rotated 270
1	3	5	7
2	4	6	8
1 & 2	3 & 4	5 & 6	7 & 8

TABLE I

Leftmost column show the failure situations that have been explicitly solved. The three other columns show the equivalent failure situations with rotated coordinated.

was evaluated using a binary search tree and the allocation scheme was implemented on a scale model of a thruster-controlled floating platform. The PWA function could be evaluated at a frequency of several kHz, well within the sampling rate of 10 Hz, and hence, freeing the computational unit for other tasks. The tracking of the generalized forces was shown to be satisfactory even if rate constraints where not included in the formulation. The method was also illustrated to perform well under single point failure situations.

As pointed out in [23] the quadratic programming formulation required by pQP may not be sufficiently flexible and powerful to incorporate all constraints and objective that may be desirable for all applications. Examples may include non-convex constraints due to non-zero minimum speed of a clutched fixed pitch propeller of an azimuth thruster, non-quadratic cost if fuel consumption is to be minimized accurately, and rate constraints due to thruster, actuator and diesel-electric power plant dynamics. As proposed by [23] a more advanced application of the explicit solution may therefore be for accurate initialization of an SQP (Sequential Quadratic Programming) solver (see e.g. [10]) that may be invoked, if necessary, to fine tune the optimal solution based on an nonlinear program that extends the pQP with the additional constraints and objectives.

REFERENCES

[1] M. Bodson, “Evaluation of optimization methods for control allocation,” *Journal of Guidance, Control, and Dynamics*, vol. 25, pp. 703–711, 2002.

[2] Y. Luo, A. Serrani, S. Yurkovich, M. W. Oppenheimer, and D. B. Doman, “Model-predictive dynamic control allocation scheme for reentry vehicles,” *Journal of Guidance, Control and Dynamics*, vol. 30, pp. 100–113, 2007.

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

[4] 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.

[5] J. Virnig and D. Bodden, “Multivariable control allocation and control law conditioning when control effectors limit,” in *Proc. AIAA Guidance, Navigation, and Control Conference*, Washington, DC, 1994, pp. 572–582.

[6] T. A. Johansen, T. I. Fossen, and P. Tøndel, “Efficient optimal constrained control allocation via multi-parametric programming,” *AIAA J. Guidance, Control and Dynamics*, vol. 28, pp. 506–515, 2005.

[7] T. A. Johansen, T. P. Fuglseth, P. Tøndel, and T. I. Fossen, “Optimal constrained control allocation in marine surface vessels with rudders,” *Control Engineering Practice*, vol. 15, pp. 457–464, 2007.

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

[9] 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, pp. 850–862, 2003.

[10] T. A. Johansen, T. I. Fossen, and S. P. Berge, “Constrained nonlinear control allocation with singularity avoidance using sequential quadratic programming,” *IEEE Trans. Control Systems Technology*, vol. 12, pp. 211–216, 2003.

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

[12] T. Gal and J. Nedoma, “Multiparametric linear programming,” *Management Science*, vol. 18, pp. 406–442, 1972.

[13] 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.

[14] P. Tøndel, T. A. Johansen, and A. Bemporad, “An algorithm for multi-parametric quadratic programming and explicit MPC solutions,” *Automatica*, vol. 39, pp. 489–497, 2003.

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

[16] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York, USA: Springer, 1999.

[17] J. Spjøtvold, P. Tøndel, and T. A. Johansen, “A continuous selection and unique polyhedral representation of solutions to convex parametric quadratic programs,” *Journal of Optimization Theory and Applications*, vol. 133, pp. 177–189, 2007.

[18] P. Tøndel, T. A. Johansen, and A. Bemporad, “Further results on multi-parametric quadratic programming,” in *Proc. 42nd IEEE Conf. on Decision and Control*, Hawaii, 2003, pp. 3173–3178.

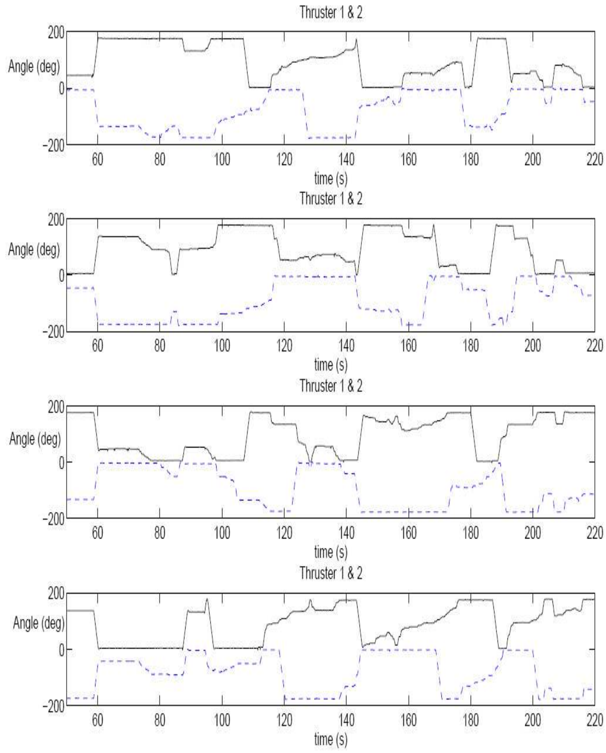
[19] C. N. Jones and M. Morari, “Multiparametric linear complementarity problems,” in *Proc. IEEE Conference on Decision and Control*, Seville, Spain, 2006, pp. 5687–5692.

[20] IMO, “IMO MSC circular resolution 645: Guidelines for vessels with dynamic positioning systems,” IMO Resolution 645, May 1994.

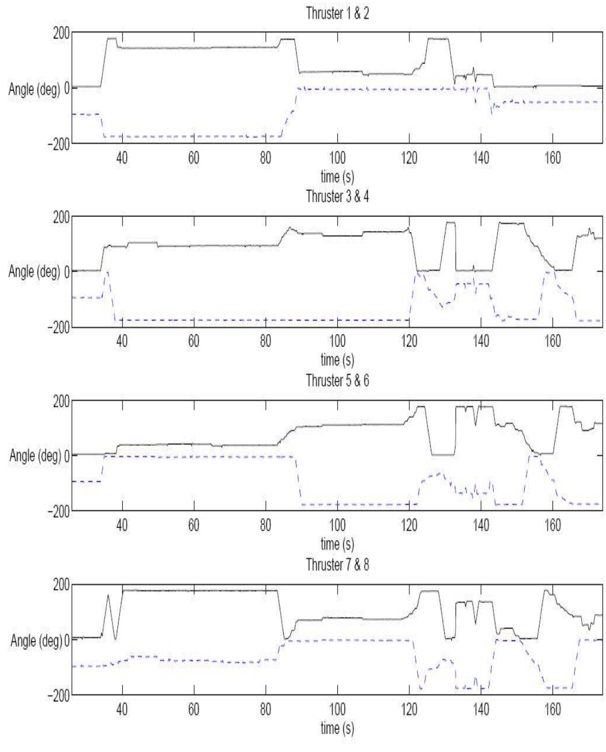
[21] International Marine Contractors Association, “IMCA M140 specification for DP capability plots,” 2000.

[22] P. Tøndel, T. A. Johansen, and A. Bemporad, “Computation of piecewise affine control via binary search tree,” *Automatica*, vol. 39, pp. 945–950, 2003.

[23] J. Leavitt, “Thrust allocation in dynamic positioning systems,” in *Proc. MTS Dynamic Positioning Conference, Houston*, 2008.

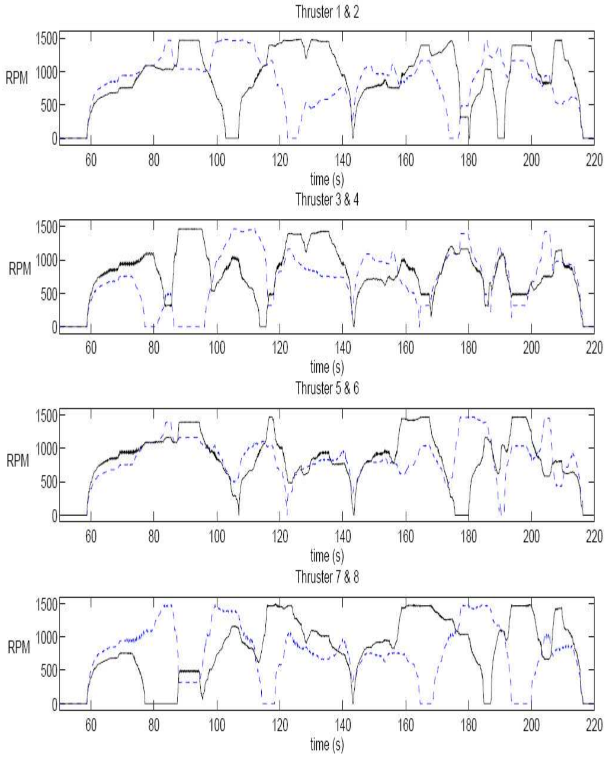


(a) Azimuth angles under nominal operation.

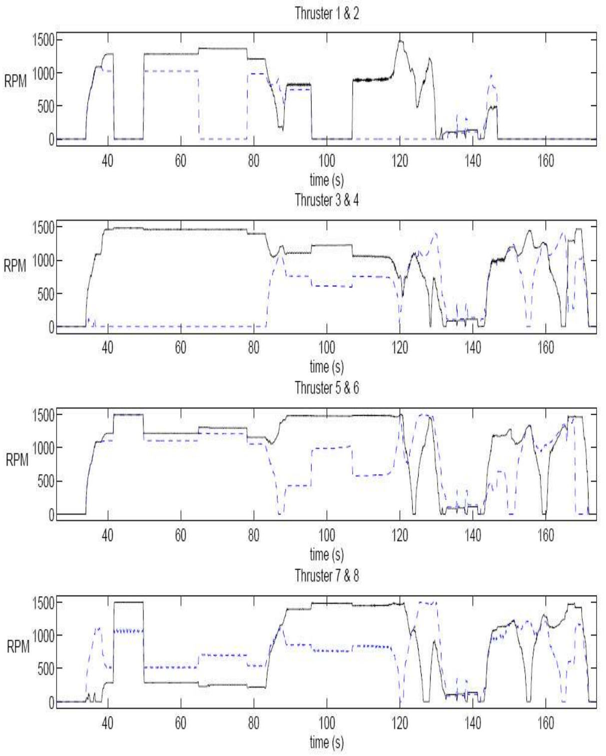


(b) Azimuth angles in failure situations.

Fig. 7. Dotted lines are azimuth angles for thrusters 1, 3, 5 and 7 and solid lines are for 2, 4, 6 and 8.



(a) RPMs under nominal operation.



(b) RPMs in failure situations.

Fig. 8. Dotted lines are RPMs for thrusters 1, 3, 5 and 7 and solid lines are for 2, 4, 6 and 8.

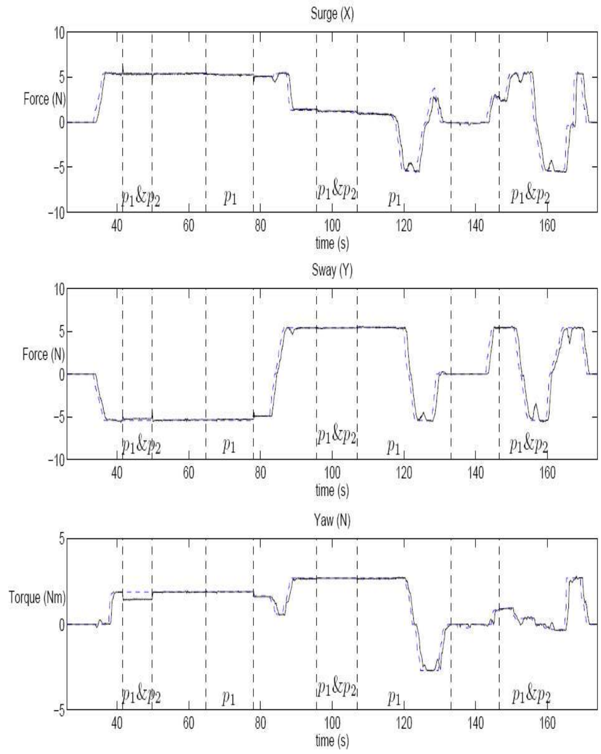


Fig. 9. Experimental results for CyberRig I; control allocation over convex ATRs. We have indicated time intervals in which either thruster 1 (p_1) or where both thruster 1 and 2 $(p_1\&p_2)$ have failed. The dotted line is the commanded and the solid line is measured.