Danish Maritime Institute: Dynamic Positioning System

JENS CHRISTIAN LARSEN, HENRIK RASMUSSEN, AND ALLAN SØRENSEN

1 Introduction

In this report, we consider the holding problem for a ship with a given arrangement of thrusters, i. e., how to ensure that a ship, floating on the surface of the ocean, stays very nearly at rest with respect to the sea bed. The aim is to determine the region of forces and torques that can be obtained by varying the directions and powers of the thrusters. In addition, for a given force and torque in this region, we would like to find, in real time, the directions and powers that should be assigned to the different thrusters.

It is important to know the holding capability when faced with the task of to keep a ship, or oil rig, at a fixed position relative to the sea bed. Unless the thrusters can be arranged such that they cancel the external forces from wind, waves, and currents, the ship/rig will drift. This can have unfortunate consequences. If an oil rig, for instance, unexpectedly starts drifting, the pipes might leak oil directly into the sea. However, knowledge of the holding capability, and the external forces in different weather conditions, makes it easier to avoid this situation.

In the section 2 we consider the problem as a discretised optimization problem and outline a procedure which give a solution to the problem. In section 3 we consider a Lagrangian formulation where we use Lagrangian multipliers to represent the side conditions. Finally in section 4 we have used Maple® to analyse a slightly different Lagrangian.

2 Solving the Problem as a Discretized Optimization Problem

In the solution described here, we assume that we are given the two components of the force F_x and F_y and the torque M_z, needed to compensate for external forces that acts on the ship, from a controller on the ship. We are not able to change the sampling period or the used strategy of this controller. We consider the following two problems:

1. The maximum holding capability problem
2. The dynamical positioning problem

2.1 The proposed solution to the problem

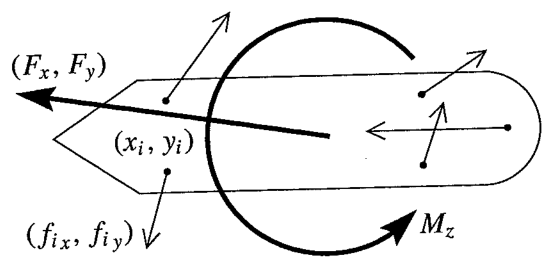


Figure 1: The required force and torque are indicated by thick arrows, while the arrangement of thrusters are indicated by thin arrows.

We call the external forces from wind, current and waves for the wind vector and denote it **W**. The vector $(F_x, F_y, M_z)^T$, describing the two components of the force and the torque needed on the boat is called the force vector and is denoted **F**. A force vector is the sum of the contributions from all the thrusters on the ship. Here we will talk about the F-space, the space of all possible vectors **F**. The relation between **W** and **F** is nonlinear, but for any given **W** there corresponds a unique vector **F**, and to each **F** it is possible to determine the possible combinations of the external forces **W**.

The first problem may be considered as a static optimization problem in the F-space. This is because the maximum holding capability problem is to determine the maximum wind that the boat is able to resist keeping a fixed position and orientation. Because of the nonlinearity between the wind vectors and the force vectors we prefer to work in F-space. That is, we want to determine the envelope E_F containing all the force vectors that can be constructed given the limitations on the thrusters. This is done by discretizing the F-space and for all possible orientations of the force vectors determine the maximum norm.

The force vector is as mentioned the sum of the contributions from all the thrusters on the ship. The ith thruster contribute with the force components f_{ix} and f_{iy} that acts on the ship on the position (x_i, y_i). Each thruster has a characteristic that depends upon the orientation of the thruster.

The determination of the maximum length of a force vector in a given direction is an optimization problem. We have as mentioned the constraints on the thrusters. Assuming convexity one can directly make use of the Kuhn-Tucker condition for an extremum and thereby obtain the maximum-force envelope E_F, see for example [1]. In the case of non-convexity one has to check the value at each extrema. The latter is what we do.

Having determined E_F one is able to consider the second problem, the dynamical positioning system. Again this is an optimization problem. If we consider the case where the thrusters are able to compensate for the applied wind, i.e. we are inside E_F, then we are looking at the following. At a given time t_i the controller compute a force vector $\mathbf{F}_i$ that is needed to control the position and orientation of the ship. We look at the related optimization problem:

$$\min \sum_{i=1}^{n} (f_{ix}^2 + f_{iy}^2)$$

given the constraints on the n thrusters etc. That is we want to find the most favorable solution with respect to the consumed energy. Again we are going to check the value at each extrema that are found by the Kuhn-Tucker conditions.

2.2 Limitations and enhancements

In the solution that we have suggested we are solving a discretized optimization problem. The reason for this is because the characteristic of the thrusters were discretized. We are able to find the finite set of local extrema by making use of the Kuhn-Tucker conditions.

Now, it is possible to obtain a continuous description of the characteristics of the thrusters. Therefore, in this case one might want to consider the continuous optimization problem instead. It is obviously not the case that it is a convex optimization problem. For example the angular-dependence of the effectiveness of the thrusters varies, due to jet-streamlines, from about 10% to 100%. To give a rough estimate on the holding capability, our discretized solution is perhaps sufficient, but with respect to the viewpoint of control theory, it should be noted, that the envelope that we have found is smaller than the actual one due to the discretization.

The strategy of control at the moment is first to keep the orientation correct then the x-coordinate of the position and finally the y-coordinate, but all these conditions are obtainable when working inside E_F. Therefore, it would of course be useful with more precise values on the maximum norms of the force vectors to make the control of the ship better.

There are several techniques to solve continuous non convex optimization problems, see for example [4]. When the problem is to determine E_F there are at least the following two useful methods: simulated annealing and dynamical annealing. These stochastic methods quickly converges to the interesting regions in the optimization space. Both techniques are roughly based on the idea of modeling the problem as a physical system where one is pumping energy into the system and then slowly pulling the energy out again.

If we have determined the maximum holding capability envelope it is possible to determine whether a given desired force vector is inside or outside the envelope. Now, there are some inertia in the system. When going from time t_i to time t_{i+1} the maximum angle the thrusters are able to turn is 6 degrees. What this means is, that if we at time t_i are at a given point in F-space then we are only able to operate within a finite subset inside the envelope at time t_{i+1}. But having determined E_F the choice of control strategy should be much easier.

3 A Lagrangian Formulation

In the previous section the holding capability of a ship with n thrusters was determined by searching through the $2n$-dimensional configuration space. Since the number of possible configurations in this phase space grows exponentially with n, for any given discretisation of the variables, this search through phase space typically takes $4-5$ hours of computing time even on a reasonably powerful machine.

The method of solution proposed in this section should be considerably faster. We write down a Lagrange function consisting of the function to be maximised and a number of extra terms representing the constraints on the variables. We show that extrema of this Lagrangian can be determined by finding extrema of n real-valued functions on the real line. The number of calculations needed to determine the extrema and their positions in phase space is proportional to n.

3.1 Mathematical formulation

Let (x_i, y_i) denote the position of the ith thruster relative to the centre of mass of the ship. Let θ_i denote its direction and f_i the power. Due to the presence of other thrusters and shielding by the hull, the absolute value of the force that a given thruster exerts on the centre of mass depends not only on the power f_i but also on the direction θ_i. In the model used at DMI, the effective force in a given direction θ_i is the product of f_i by an efficiency factor, $\eta_i(\theta_i)$.

Given an external force and torque acting on the centre of mass of ship, we would like to produce an opposing force and an opposing torque by means of the thrusters. Let (F_x, F_y) denote the desired force and M_z the desired torque. Our task is then to find f_i and θ_i such that,

$$F_x = \sum_i f_i\, \eta_i(\theta_i)\, \cos\theta_i,$$

$$F_y = \sum_i f_i\, \eta_i(\theta_i)\, \sin\theta_i, \tag{1}$$

$$M_z = \sum_i f_i\, \big(-y_i\, \cos\theta_i + x_i\, \sin\theta_i\big).$$

Since the thrusters are only capable of finite power, we have in addition n constraints in the form,

$$f_{\min} \le f_i \le f_{\max}, \quad i = 1, \dots, n. \tag{2}$$

The lower bound $f_{\min}$ will usually be negative, as the thrusters are capable of reversing.

In this report, we define a ship's holding capability as the collection of 'demanded' forces and torques (F_x, F_y, M_z) for which the preceding system of equations admit a solution. The determination of the holding capability is simplified by the fact that the holding capability is a star-shaped set. That is, if (F_x, F_y, M_z) is a point within the ship's holding capability, then every point on a line segment from the origin $(0, 0, 0)$ to (F_x, F_y, M_z) is also within the ship's holding capability. This is a simple consequence of the right-hand-sides of Eqs. (1) being linear in $(f_1, \dots, f_n)$. Hence, there is no loss of generality in restricting our attention to values of (F_x, F_y, M_z) that lie on a given straight line passing through the origin.

For each pair of real constants α and β, consider the lines in the space (F_x, F_y, M_z) parameterised by F_x as follows,

$$F_y = \alpha\, F_x, \qquad M_z = \beta\, F_x. \tag{3}$$

(For the special case where $F_x = 0$, consider instead a linear relationship between F_y and M_z). By Eqs. (1), these equations can be written as follows,

$$\sum_i f_i\, \eta_i(\theta_i)\, \big(\sin\theta_i - \alpha\, \cos\theta_i\big) = 0, \tag{4}$$

$$\sum_i f_i\, \eta_i(\theta_i)\, \big(-y_i\, \cos\theta_i + x_i\, \sin\theta_i - \beta\, \cos\theta_i\big) = 0. \tag{5}$$

Using the simpler notation,

$$\begin{aligned} g_i(\theta_i) &= \eta_i(\theta_i)\, \big(\sin\theta_i - \alpha\, \cos\theta_i\big), \\ h_i(\theta_i) &= \eta_i(\theta_i)\, \big(-y_i\, \cos\theta_i + x_i\, \sin\theta_i - \beta\, \cos\theta_i\big), \\ k_i(\theta_i) &= \eta_i(\theta_i)\, \cos\theta_i, \end{aligned} \tag{6}$$

we get

$$F_x = \sum_i f_i\, k_i,$$
$$F_y - \alpha\, F_x = \sum_i f_i\, g_i,$$
$$M_z - \beta\, F_x = \sum_i f_i\, h_i.$$

Note that g_i, h_i, and k_i are known functions of θ_i, while the f_i are variables. The full statement of our problem is then: find the global maxima and minima of

$$F_x = \sum_i f_i\, k_i(\theta_i), \tag{7}$$

subject to the constraints

$$\sum_i f_i\, g_i(\theta_i) = 0,$$
$$\sum_i f_i\, h_i(\theta_i) = 0, \tag{8}$$
$$f_{\min} \le f_i \le f_{\max}, \quad i = 1, \ldots, n.$$

3.2 Local extrema

The optimization of F_x in (7) subject to the constraints in Eqs. (8) is of a classical type and has a well-known method of solution (e.g., [2], [5]). Functions describing the constraints are multiplied by undetermined constants - the Lagrange multipliers - and added to the function to be optimized, F_x. This procedure yields the following Lagrangian function,

$$L(f_1, \ldots, f_n, \theta_1, \ldots \theta_n) = F_x + \lambda_1\,(F_y - \alpha\, F_x) + \lambda_2\,(M_z - \beta\, F_x) + \sum_i \mu_i\,(f_i - f_{\min}) + \sum_i \kappa_i\,(f_{\max} - f_i),$$

where λ_1, λ_2, the μ_i and the κ_i are the undetermined Lagrangian multipliers. The advantage of optimizing the Lagrangian L instead of the original function F_x is that an optimum of the former automatically satisfies the constraints.

The μ_i and the κ_i are constraints on the domain and they are set to zero unless $f_i = f_{\max}$ or $f_i = f_{\min}$, respectively. However, λ_1 and λ_2 are not set to zero at the boundary.

Using the functions k_i, g_i and h_i that were defined above, the Lagrangian may be written as follows,

$$L = \sum_i f_i\,\left(k_i + \lambda_1\, g_i + \lambda_2\, h_i + \mu_i - \kappa_i\right) + \sum_i \left(-\mu_i\, f_{\min} + \kappa_i\, f_{\max}\right). \tag{9}$$

The first-order condition for having an optimum is that the gradient of L with respect to $(f_1, \ldots, f_n, \theta_1, \ldots, \theta_n)$ vanishes. This condition gives $2n$ equations to satisfy, one for each i,

$$k_i + \lambda_1\, g_i + \lambda_2\, h_i + \mu_i - \kappa_i = 0, \tag{10}$$
$$k_i' + \lambda_1\, g_i' + \lambda_2\, h_i' = 0. \tag{11}$$

We now consider the solution in the cases where n, $n-2 \ldots$, and so on, of the thrusters are set at either minimum or maximum power.

Case I: all n thrusters on full power

When all n thrusters are set at either minimum or maximum power, Eq. (10) contains either a μ_i or a κ_i for each $i = 1, \ldots n$. For any given choice of the directions $\{\theta_i\}$, it is always possible to choose this constant such that Eq. (10) holds. Equation (10) then puts no restriction on θ_i, and we need only consider Eq. (11).

To determine whether a given solution $\{\theta_i\}_{i=1,..n}$ of Eq. (11), for each i, is indeed a local extremum for F_x, one can consider higher-order derivatives of the Lagrangian L. Sufficient and necessary conditions for local extrema can be found in [2]; note that we consider only necessary conditions here. If this strategy of solution were to be implemented numerically, we would have to solve, for some choice of λ_1 and λ_2 (see the comment in the summary), the n equations represented by Eq. (11). However, the number of calculations needed to solve these n equations grows faster than n. Since the efficiency factor η_i of the ith thruster is influenced by the relative positions of all other thrusters, the functions k_i, g_i and k_i depend on n. Consequently, in the general case, the number of solutions for $\{\theta_i\}_{i=1,..n}$ in Eq. (11) is of order n^2 rather than n.

However, it is possible to reduce the complexity to one of order n by finding θ_i, in the case of a maximum, as follows,

$$k_i(\theta_i) + \lambda_1\, g_i(\theta_i) + \lambda_2\, h_i(\theta_i) = \max_{0 \le \theta < 2\pi}\left[k_i(\theta) + \lambda_1\, g_i(\theta) + \lambda_2\, h_i(\theta)\right]. \tag{12}$$

The algorithms used for finding extrema solve this problem in a time that depends only on the discretisation chosen and not on the number of local maxima in the functions f_i, g_i and h_i. We have thus reduced the original problem from one that grows exponentially with the number n of thrusters to one that grows linearly with the number of thrusters.

Finally, note that λ_1 and λ_2 are undetermined. This shows that the directions $\{\theta_i\}$ yielding extrema of F_x form a two-dimensional surface in the n-dimensional space of all possible directions. The surface is two-dimensional because we have imposed two constraints. These two constraints are represented by the parameters α and β in Eqs. (3).

Case II: exactly n − 1 thrusters on full power

Let us suppose that the jth thruster is not on full power. That is,

$$f_{\min} < f_j < f_{\max}, \tag{13}$$

where it is important that strict inequality holds on both sides. The constraints associated with μ_j and κ_j are now passive, and Eqs. (10) and (11) become, for $i = j$,

$$\begin{aligned} k_j + \lambda_1 g_j + \lambda_2 h_j &= 0, \\ k'_j + \lambda_1 g'_j + \lambda_2 h'_j &= 0. \end{aligned} \tag{14}$$

Once θ_j is given, the k_j, k'_j, g_j ... and so on appearing in Eqs. (14) are determined. We then have two linear equations with two unknowns, λ_1 and λ_2. In general, this system of equations can be inverted and thereby λ_1 and λ_2 are determined uniquely as functions of θ_j. The exception is when the determinant of the system is zero. Then one or both equations are negligible (either one is linearly proportional to the other or both are trivially satisfied; we do not discuss these cases in detail, as they are non-generic, but they represent no difficulty in principle). The solutions for all other values of i are determined as before, from Eq. (11), with the caveat that λ_1 and λ_2 are now the solutions of Eqs. (14). As in Case I, we obtain a two-dimensional collection of solutions, but now parameterised by (θ_j, f_j).

Note that the θ_i with $i \neq j$ are determined indirectly by θ_j through λ_1 and λ_2. Thus, depending on which thruster is not on full power, different solutions appear.

Case III: exactly n − 2 thrusters on full power

In this case, two thrusters are not on full power. If these two thrusters are number j_1 and j_2, with $j_1 \neq j_2$, then an extremum exists only if we can solve Eqs. (14) for both $j = j_1$ and $j = j_2$. If we consider first the case of $j = j_1$, then the possible choices of λ_1 and λ_2 form a closed curve in the plane, parameterised by θ_{j1}. Similarly, if we consider the case $j = j_2$, then we get another closed curve, this time parameterised by θ_{j2}. Since we have to be on both curves at the same time, the admissible values of λ_1 and λ_2 are the points of intersection. There may be any number between zero and infinity of such intersection points. (The latter happens when the efficiency functions η_{j1} and η_{j2} are identical).

Case IV: n − 3 or less thrusters on full power

Now we have to satisfy Eqs. (14) for three distinct values of j. Since we have six equations, but only five variables (the three directions and the two Lagrange multipliers), this is impossible in the generic case. Hence, we obtain a 'Sailors Lemma',

For a ship with n thrusters, where n ≥ 3, at least n − 2 thrusters are needed to achieve maximal force and torque at the same time.

But the practical relevance of this lemma is questionable, to say the least.

3.3 Summary of method

The method proposed in the previous section for solving the holding problem can be summarised as follows:

1. Decide which α and β to use in Eqs. (3).
2. Given the efficiency factors $\eta_i(\theta_i)$, for $i = 1, \ldots n$, construct the functions f_i, g_i and h_i in Eqs. (6).
3. Determine the local extrema as described under cases I–III in the previous section.
4. Compare these local extrema to see which are global minima or maxima. The corresponding directions $\{\theta_i\}$ and power allocations $\{f_i\}$ are then solutions to the holding problem for this particular choice of α and β.

This procedure is repeated for as many values of α and β as desired.

As for the third point, it is important to note the following. If the functions f_i, g_i and h_i are all smooth, then it is sufficient, in case I, to optimize for only one choice of λ_1 and λ_2. Geometrically, the maximum of F_x occurs at a point where the gradient of F_x is orthogonal to the surface on which the two constraints are satisfied. The collection of points at which this conditions is satisfied is a surface parameterised by λ_1 and λ_2. Since the gradient of F_x is orthogonal to this surface, F_x is constant on the surface and the value of F_x at any point on the surface is the maximum. Similar simplifications are possible for cases II and III.

4 A Maple® Program

4.1 Statement of the first problem

We are given a floating oil rig influenced by wind and current. There are two components of the total force on the oil rig, the x component Fx, and the y component Fy, in a rectangular coordinate system xy. This force can change the position of the oil rig. In addition to this there is a moment Mz which can change the direction of the vessel.

To compensate for the changes the force and torque can give we have n thrusters at positions $(x1, y1), \ldots, (xn, yn)$ on the ship. This means we want to satisfy the equations

$$Fx = \sum_{i=1}^{n} fx_i$$

$$Fy = \sum_{i=1}^{n} fy_i$$

$$Mz = \sum_{i=1}^{n} (fx_i\, y_i - fy_i\, x_i)$$

We formulate this as an optimzation problem: Minimize the energy function

$$E = \sum_{i=1}^{n} (fx_i^2 + fy_i^2)$$

subject to the formulas above.

4.2 Solution to the first problem

We start by solving this problem. For the sake of simplicity we let $n = 4$.

First we solve for fx_1 in the first equation above and for fy_1 in the second. Finally we isolate fx_2 from the third equation.


```
>   restart;
>   fx1:=Fx-sum(fx[i],i=3..4)-fx2;
```

$$fx1 := Fx - fx_3 - fx_4 - fx2$$

```
>   fy1:=Fy-sum(fy[i],i=2..4);
```

$$fy1 := Fy - fy_2 - fy_3 - fy_4$$

```
>   eq:=Mz-sum(fx[i]*y[i]-fy[i]*x[i],i=3..4)+fy[2]
>   *x[2]+fy1*x[1]-fx1*y[1]-fx2*y[2];
```

$$eq := Mz - fx_3\, y_3 + fy_3\, x_3 - fx_4\, y_4 + fy_4\, x_4 + fy_2\, x_2 + (Fy - fy_2 - fy_3 - fy_4)\, x_1 - (Fx - fx_3 - fx_4 - fx2)\, y_1 - fx2\, y_2$$

```
>   fx2:=solve(eq,fx2);
```

$$fx2 := -(-Mz + fx_3\, y_3 - fy_3\, x_3 + fx_4\, y_4 - fy_4\, x_4 - fy_2\, x_2 - x_1\, Fy + x_1\, fy_2 + x_1\, fy_3 + x_1\, fy_4 + y_1\, Fx - y_1\, fx_3 - y_1\, fx_4) \Big/ (-y_1 + y_2)$$

Now the energy function E that we wish to minimize is

```
>   E:=sum(fx[i]^2+fy[i]^2,i=3..4)
+fx1^2+fx2^2+fy1^2;
```

$$E := fx_3{}^2 + fy_3{}^2 + fx_4{}^2 + fy_4{}^2 + (Fx - fx_3 - fx_4 + (-Mz + fx_3\, y_3 - fy_3\, x_3 + fx_4\, y_4 - fy_4\, x_4 - fy_2\, x_2 - x_1\, Fy + x_1\, fy_2 + x_1\, fy_3 + x_1\, fy_4 + y_1\, Fx - y_1\, fx_3 - y_1\, fx_4) \Big/ (-y_1 + y_2))^2 + (-Mz + fx_3\, y_3 - fy_3\, x_3 + fx_4\, y_4 - fy_4\, x_4 - fy_2\, x_2 - x_1\, Fy + x_1\, fy_2 + x_1\, fy_3 + x_1\, fy_4 + y_1\, Fx - y_1\, fx_3 - y_1\, fx_4)^2 \Big/ (-y_1 + y_2)^2 + (Fy - fy_2 - fy_3 - fy_4)^2$$

To find the minimum we compute the following partial derivatives

```
>   eq1:=diff(E,fx[3]):
>   eq2:=diff(E,fx[4]):
>   eq3:=diff(E,fy[2]):
>   eq4:=diff(E,fy[3]):
>   eq5:=diff(E,fy[4]):
```

and solve for the remaining variables fx_3, fx_4, fy_2, fy_3, fy_4.

```
>   sol2:=solve({eq1,eq2,eq3,eq4,eq5}, {fx[3],fx[4],
fy[2],fy[3],fy[4]}):
```

So for small values of Fx, Fy and Mz we have found the forces the thrusters should give.

Here are the formulas for fx_3, fx_4, fy_3, and fy_4:

```
>   fxx3:=subs(sol2,fx[3]);fxx4:=subs(sol2,fx[4]);
```

$$fxx3 := (3\,x_2^2\,Fx - x_2\,y_2\,Fy - 2\,x_2\,x_1\,Fx - 2\,x_2\,Fx\,x_3 - x_2\,y_4\,Fy + 3\,x_2\,y_3\,Fy - x_2\,y_1\,Fy - 2\,x_2\,Fx\,x_4 - Fx\,y_2\,y_3 + 3\,Mz\,y_3 - y_1\,Fx\,y_3 + Fx\,x_3^2 + x_1^2\,Fx - Mz\,y_2 - Mz\,y_4 + Fx\,y_2^2 - Mz\,y_1 + y_1^2\,Fx - Fx\,y_3\,y_4 + Fx\,y_4^2 + Fx\,x_4^2) \Big/ (3\,y_2^2 - 2\,y_2\,y_4 - 2\,y_1\,y_2 - 2\,y_3\,y_2 - 2\,y_3\,y_1 + 12\,x_2^2 + 3\,y_4^2 + 4\,x_3^2 + 3\,y_1^2 + 4\,x_4^2 - 2\,y_3\,y_4 - 2\,y_1\,y_4 - 8\,x_2\,x_4 + 3\,y_3^2 + 4\,x_1^2 - 8\,x_2\,x_3 - 8\,x_1\,x_2)$$

$$fxx4 := (3\,x_2^2\,Fx - 2\,x_2\,Fx\,x_3 + 3\,x_2\,y_4\,Fy - x_2\,y_3\,Fy - x_2\,y_2\,Fy - x_2\,y_1\,Fy - 2\,x_2\,x_1\,Fx - 2\,x_2\,Fx\,x_4 - Mz\,y_3 + Fx\,x_3^2 + Fx\,y_3^2 + x_1^2\,Fx - Mz\,y_2 + 3\,Mz\,y_4 + Fx\,y_2^2 - Mz\,y_1 + y_1^2\,Fx - Fx\,y_2\,y_4 - Fx\,y_3\,y_4 - y_1\,Fx\,y_4 + Fx\,x_4^2) \Big/ (3\,y_2^2 - 2\,y_2\,y_4 - 2\,y_1\,y_2 - 2\,y_3\,y_2 - 2\,y_3\,y_1 + 12\,x_2^2 + 3\,y_4^2 + 4\,x_3^2 + 3\,y_1^2 + 4\,x_4^2 - 2\,y_3\,y_4 - 2\,y_1\,y_4 - 8\,x_2\,x_4 + 3\,y_3^2 + 4\,x_1^2 - 8\,x_2\,x_3 - 8\,x_1\,x_2)$$

```
> fyy3:=subs(sol2,fy[3]); fyy4:=subs(sol2,fy[4]);
```

$$fyy3 := -(-Fx\,y_2\,x_3 + Fx\,y_2\,x_2 - 4\,Mz\,x_2 - 4\,Fy\,x_2^2 + y_3\,Fx\,x_2 - y_3\,Fx\,x_3 + 4\,Mz\,x_3 - y_1\,Fx\,x_3 + 4\,x_2\,x_3\,Fy + y_1\,Fx\,x_2 - y_4\,Fx\,x_3 + y_4\,Fx\,x_2) \Big/ (3\,y_2^2 - 2\,y_2\,y_4 - 2\,y_1\,y_2 - 2\,y_3\,y_2 - 2\,y_3\,y_1 + 12\,x_2^2 + 3\,y_4^2 + 4\,x_3^2 + 3\,y_1^2 + 4\,x_4^2 - 2\,y_3\,y_4 - 2\,y_1\,y_4 - 8\,x_2\,x_4 + 3\,y_3^2 + 4\,x_1^2 - 8\,x_2\,x_3 - 8\,x_1\,x_2)$$

$$fyy4 := -(-4\,Fy\,x_2^2 + y_4\,Fx\,x_2 + y_1\,Fx\,x_2 + y_3\,Fx\,x_2 + Fx\,y_2\,x_2 + 4\,x_2\,Fy\,x_4 - 4\,Mz\,x_2 - y_3\,Fx\,x_4 + 4\,Mz\,x_4 - y_4\,Fx\,x_4 - Fx\,y_2\,x_4 - y_1\,Fx\,x_4) \Big/ (3\,y_2^2 - 2\,y_2\,y_4 - 2\,y_1\,y_2 - 2\,y_3\,y_2 - 2\,y_3\,y_1 + 12\,x_2^2 + 3\,y_4^2 + 4\,x_3^2 + 3\,y_1^2 + 4\,x_4^2 - 2\,y_3\,y_4 - 2\,y_1\,y_4 - 8\,x_2\,x_4 + 3\,y_3^2 + 4\,x_1^2 - 8\,x_2\,x_3 - 8\,x_1\,x_2)$$

```
> fxx1:=subs(sol2,fx1):
> fxx2:=subs(sol2,fx2):
> fyy1:=subs(sol2,fy1):
> fyy2:=subs(sol2,fy[2]):
```

We can easily substitute values for the constants by using the MAPLE command subs:

```
> subs({Mz=4,Fx=13,Fy=5,x[1]=0,x[2]=2,x[3]=2,
x[4]=3,y[1]=1,y[2]=1,y[3]=2,y[4]=2},fyy4);
```

$$\frac{11}{12}$$

4.3 Numerical experiment

Here we perform a numerical experiment by letting Fx, Fy and Mz increase along a line through the origin and then solve for the values of the thruster forces. We have found the solution to the control problem above for small wind speeds. However

at some point one of the motors will reach its maximum capability and then we have a different control problem.

We shall impose the natural restriction that each motor has only a limited amount of power according to inequalities

$$-g_i(fx_i) \leq fy_i \leq g_i(fx_i)$$

where the g_i are smooth functions. For the sake of simplicity we set

```
> x[1]:=0:x[2]:=2:x[3]:=2:x[4]:=1:y[1]:=0.2:
y[2]:=1:y[3]:=2:y[4]:=2:
```

The solution to this problem can be found by using the Kuhn Tucker Theorem. This theorem says that if we want to minimize $E(fx, fy)$ subject to the constraints

$$0 = h1(fx, fy) = Fx - \sum_{i=1}^{n} fx_i$$

$$0 = h2(fx, fy) = Fy - \sum_{i=1}^{n} fy_i$$

$$0 = h3(fx, fy) = Mz - \sum_{i=1}^{n} (fx_i\, y_i - fy_i\, x_i)$$

and

$$ki(fx, fy) = gi(fx_4) - fy_4 \leq 0$$

we have to solve the equation

$$\text{grad}\, E(fx, fy) + \sum_{i=1}^{n} \lambda_i\ \text{grad\_}hi(fx, fy) + \mu_1\ \text{grad\_}ki(fx, fy) \qquad (*)$$

where λ_1, λ_2, λ_3, and μ_1 are real constants to be determined. This is done by the following MAPLE code.

```
> B:=matrix(8,4,[-1,0,-y[1],0, -1,0,-y[2],0,
-1,0,-y[3],0, -1,0,-y[4],H, 0,-1,x[1],0,
0,-1,x[2],0, 0,-1,x[3],0, 0,-1,x[4],-1]);
```

$$B := \begin{bmatrix} -1 & 0 & -.2 & 0 \\ -1 & 0 & -1 & 0 \\ -1 & 0 & -2 & 0 \\ -1 & 0 & -2 & H \\ 0 & -1 & 0 & 0 \\ 0 & -1 & 2 & 0 \\ 0 & -1 & 2 & 0 \\ 0 & -1 & 1 & -1 \end{bmatrix}$$

In the first three columns of B we have grad h_1, grad h_2, grad h_3 and in the last we have grad k_i. We have let $H = g_i'(fx_4)$.

```
>  with(linalg):
```

```
Warning, new definition for norm

Warning, new definition for trace
```

The Kuhn Tucker equations (*) are

```
>  eqn:={seq(evalm(B&*(vector([l1,l2,l3,m1])))[i]
>  +2*vector([fx1,fx2,fx[3],fx[4],fy1, fy[2],fy[3],
fy[4]])[i]=0,i=1..8)};
```

$$
\begin{aligned}
eqn := \{&-l2 + 2\,l3 + 2 fy_3 = 0,\ -l2 + 2\,l3 + 2 fy_2 = 0,\ -l1 - 2\,l3 + 2 fx_3 = 0,\\
&-l1 - 2\,l3 + H\,m1 + 2 fx_4 = 0,\\
&-l1 - .2\,l3 + 2.500000000\,Fx + 2.500000000\,fx_3 + 2.500000000\,fx_4 - 2.500000000\,Mz\\
&- 5.000000000\,fy_3 - 2.500000000\,fy_4 - 5.000000000\,fy_2 = 0,\\
&- l2 + 2\,Fy - 2 fy_2 - 2 fy_3 - 2 fy_4 = 0,\ -l2 + l3 - m1 + 2 fy_4 = 0,\\
&- l1 - l3 + 2.500000000\,Mz - 4.500000000\,fx_3 + 5.000000000\,fy_3 - 4.500000000\,fx_4\\
&+ 2.500000000\,fy_4 + 5.000000000\,fy_2 - .5000000000\,Fx = 0\}
\end{aligned}
$$

```
>  fy[4]:=H1;
```

$$fy_4 := H1$$

Later we will let $H1 = g_i(fx_4)$. For the case of one motor on maximum capability the solution is

```
>  sol:=solve(eqn,{l1,l2,l3,m1,fx[3],fx[4],fy[2],
fy[3]});
```

$$
\begin{aligned}
sol := \{&\\
&fy_2 = .1428571429\,\frac{-100.\,Mz + 114.\,Fy - 214.\,H1 + 130.\,Fx + 35.\,H\,Fy - 105.\,H\,H1}{106. + 5.\,H},\\
&fy_3 = .1428571429\,\frac{-100.\,Mz + 114.\,Fy - 214.\,H1 + 130.\,Fx + 35.\,H\,Fy - 105.\,H\,H1}{106. + 5.\,H},\\
&fx_4 = -.07142857143(-210.\,Mz - 280.\,Fy + 70.\,H1 - 98.\,Fx - 422.\,H\,Fy\\
&+1313.\,H\,H1 - 75.\,Mz\,H + 80.\,Fx\,H)/(106. + 5.\,H),\\
&l3 = -1.428571429\,\frac{-30.\,Mz - 40.\,Fy + 10.\,H1 + 39.\,Fx + 7.\,H\,Fy - 28.\,H\,H1}{106. + 5.\,H},\\
&l2 = -.2857142857\,\frac{-200.\,Mz - 514.\,Fy + 314.\,H1 + 260.\,Fx + 35.\,H\,Fy - 175.\,H\,H1}{106. + 5.\,H},\\
&fx_3 = .07142857143(210.\,Mz + 280.\,Fy - 70.\,H1 + 98.\,Fx - 206.\,H\,Fy + 699.\,H\,H1\\
&-25.\,Mz\,H + 50.\,Fx\,H)/(106. + 5.\,H),\\
&l1 = .1428571429(-390.\,Mz - 520.\,Fy + 130.\,H1 + 878.\,Fx - 66.\,H\,Fy + 139.\,H\,H1\\
&-25.\,Mz\,H + 50.\,Fx\,H)/(106. + 5.\,H),\\
&m1 = .2857142857\,\frac{-50.\,Mz - 314.\,Fy + 1006.\,H1 + 65.\,Fx}{106. + 5.\,H}\}
\end{aligned}
$$


```
>  fX3:=subs(sol,fx[3]):
```

In the MAPLE program below we shall need all the motor forces, so we have to assign values to all the fx_i, fy_i.

```
>  fY4:=H1;
```

$$fY4 := H1$$

```
>  fX1:=subs(sol,fx1):
>  fY1:=subs(sol,fy1):
>  fX2:=subs(sol,fx2):
>  fY2:=subs(sol,fy[2]):
>  fY3:=subs(sol,fy[3]):
```

Just to consider an explicit example we shall use the motor constraint function

```
>  gi:=x->sqrt(20^2-x^2/2);
```

$$gi := x \to \sqrt{400 - \frac{1}{2}x^2}$$

```
>  H1:=gi(fx[4]);
```

$$H1 := \frac{1}{2}\sqrt{1600 - 2 fx_4^2}$$

```
>  H:=diff(H1,fx[4]);
```

$$H := -\frac{fx_4}{\sqrt{1600 - 2 fx_4^2}}$$

To find the solution to the optimization problem we solve the following equation for fx_4. Once this is solved we can insert in the formulas for the fx_i and fy_i.

```
>  sol1:=fx[4]=subs(sol,fx[4]);
```

$$
\begin{aligned}
sol1 := fx_4 = &-.07142857143(-210.\,Mz - 280.\,Fy + 35.00000000\sqrt{\%1} - 98.\,Fx\\
&+ 422.\,\frac{fx_4\,Fy}{\sqrt{\%1}} - 656.5000000 fx_4 + 75.\,\frac{Mz\,fx_4}{\sqrt{\%1}} - 80.\,\frac{Fx\,fx_4}{\sqrt{\%1}}) \Big/ (106. - 5.\,\frac{fx_4}{\sqrt{\%1}})\\
\%1 := &\ 1600 - 2 fx_4^2
\end{aligned}
$$

Here we plot the right hand side of the above equation for a particular choice of Fx, Fy and Mz. We display it below. We can solve the equation $sol1$ numerically using the MAPLE command fsolve once we assign values to Fx, Fy, and Mz. This is done below.

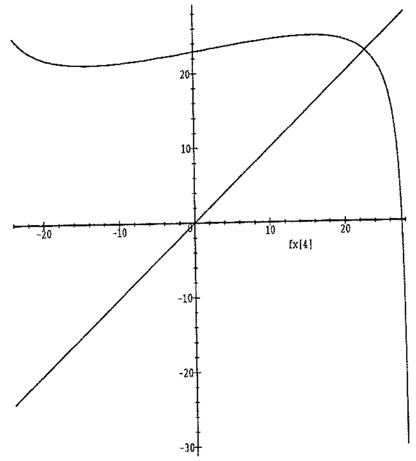
```
>  plot1:=plot(rhs(subs({Fx=11*10,Fy=11*5,
Mz=11*4},sol1)),fx[4]=-24..28):
>  eq12:=subs({Fx=13*10,Fy=13*5,Mz=13*4},sol1);
fX4:=fsolve(eq12,fx[4],20..28);
```

$$eq12 := fx_4 = -.07142857143 \left(-41860. + 35.00000000\sqrt{1600 - 2fx_4^2} + 20930.\,\frac{fx_4}{\sqrt{1600 - 2fx_4^2}} - 656.5000000\,fx_4 \right) \Big/ \left(106. - 5.\,\frac{fx_4}{\sqrt{1600 - 2fx_4^2}} \right)$$

$$fX4 := 23.85218616$$

We can find the solution to equation *sol1* graphically by plotting the right hand side and left hand side of the equation. The point of intersection gives the solution.

```
> with(plots):
> plot2:=plot(y,y=-24..28):
> display(plot1,plot2);
```



We can see that the plot agrees well with the numerical solution $fx_4 = 23.85$ that we found above using fsolve.

4.4 Output of the MAPLE program

We have chosen to omit the MAPLE program used to perform the numerical experiment. Below we display the output of the program, showing the motor forces found by the solution formulas above as we increase (Fx, Fy, Mz) along the line

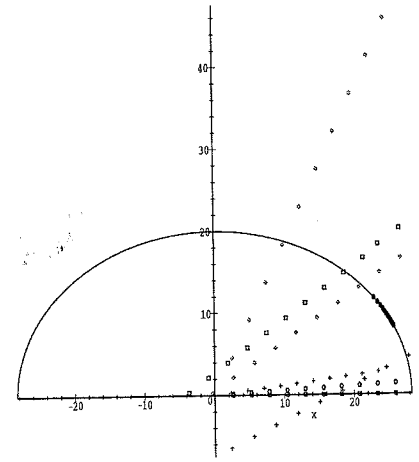
$$t(10, 5, 4), \quad t \in \mathbb{R},$$

through the origin.

```
> x0:=evalf(sqrt(400*2));
```

$$x0 := 28.28427124$$


```
> plot1:=plot(gi(x),x=-x0..x0):
> display(plot3,plot1,plot2,plot4,plot5);
```



Motor 1: diamond, Motor 2: cross, Motor 3: box, and Motor 4 (reaching the boundary of the constraints): circle.

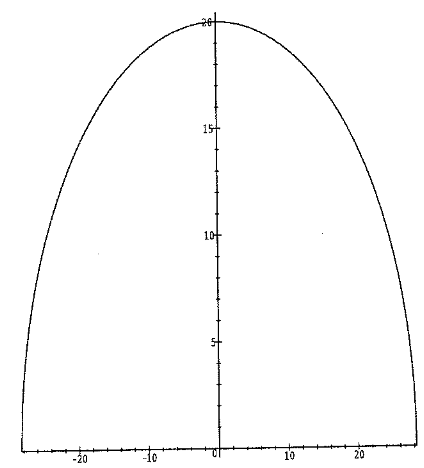
4.5 Maximum capability

In the following we shall determine the maximum capability of the oil rig, that is we shall determine the values of (Fx, Fy, Mz) that we have motor power to compensate for.

We shall use the motor constraint function

$$gi(fx_4) = \sqrt{400 - \frac{1}{2}x^2},$$

and we plot it below.



To find the maximum capability of the oil rig we discretize the values of the motor forces fx_i, fy_i and plug into the formulas for Fx, Fy, Mz. Finally we plot the resulting (Fx, Fy, Mz) values.

```
> L1:=[evalf(seq(seq(seq(seq([i*(28/3),j*(28/4),
k*(28/4),l*(28/4)],i=-3..3),j=-3..3),k=-3..3),
l=-3..3))]:
```

```
> x->(gi(x[1]),gi(x[2]),gi(x[3]),gi(x[4]),x[1],
x[2],x[3],x[4]);
```

$$x \to (\mathrm{gi}(x_1),\ \mathrm{gi}(x_2),\ \mathrm{gi}(x_3),\ \mathrm{gi}(x_4),\ x_1,\ x_2,\ x_3,\ x_4)$$

```
> L2:=map(x->[gi(x[1]),gi(x[2]),gi(x[3]),gi(x[4]),
x[1],x[2],x[3],x[4]],L1):
```

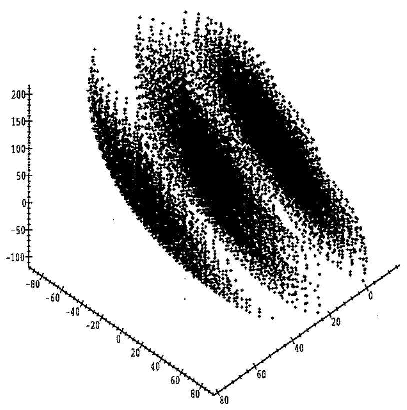
```
> x[1]:=0:x[2]:=2:x[3]:=2:x[4]:=2:
y[1]:=1:y[2]:=1:y[3]:=2:y[4]:=2:
```

```
> L3:=evalf(map(z->[sum(z[i],i=1..4),
sum(z[j],j=5..8),sum(z[k]*x[k]-z[k+4]*y[k],
k=1..4)],L2)):
```

```
> with(plots):
```

The following is a plot of the values we have motor power to compensate for in the (Fx, Fy, Mz) coordinate system. Above we only discretized part of the boundary, just to show the idea involved.

```
> pointplot3d(L3,style=point,symbol=cross,
color=blue,axes=FRAMED);
```



4.6 The case with two motors on maximum capability

When two motors are on maximum capability the Kuhn Tucker equation becomes

$$\mathrm{grad}\, E(fx,\ fy) + \sum_{i=1}^{n} li\ \mathrm{grad\_}hi(fx,\ fy) + \sum_{i=1}^{n} mi\ \mathrm{grad\_}ki(fx,\ fy) = 0,$$

and this is solved below for $n = 4$. We use the notation

$$Hi = \frac{\partial}{\partial fx_i}\,\mathrm{ki}(fx,\ fy)$$
$$Ki = ki(fx,\ fy)$$

```
> restart;
```

The first column in the matrix B below is grad $h1$, the second is grad $h2$, the third is grad $h3$ and the last four are grad ki.

```
> B:=matrix(8,7,[-1,0,x[1],0,0,0,H1,
-1,0,x[2],0,0,H2,0, -1,0,x[3],0,H3,0,0,
-1,0,x[4],H4,0,0,0, 0,-1,-y[1],0,0,0,-1,
0,-1,-y[2],0,0,-1,0, 0,-1,-y[3],0,-1,0,0,
0,-1,-y[4],-1,0,0,0]);
```

$$B := \begin{bmatrix} -1 & 0 & x_1 & 0 & 0 & 0 & H1 \\ -1 & 0 & x_2 & 0 & 0 & H2 & 0 \\ -1 & 0 & x_3 & 0 & H3 & 0 & 0 \\ -1 & 0 & x_4 & H4 & 0 & 0 & 0 \\ 0 & -1 & -y_1 & 0 & 0 & 0 & -1 \\ 0 & -1 & -y_2 & 0 & 0 & -1 & 0 \\ 0 & -1 & -y_3 & 0 & -1 & 0 & 0 \\ 0 & -1 & -y_4 & -1 & 0 & 0 & 0 \end{bmatrix}$$

```
> with(linalg):
```

```
Warning, new definition for norm

Warning, new definition for trace
```

The Kuhn Tucker equations are

```
> eqn:={seq(evalm(B&*(vector([l1,l2,l3,m1,m2,0,0])
))[i]+2*vector([fx1,fx2,fx[3],fx[4],fy1,fy[2],
fy[3],fy[4]])[i]=0,i=1..8)};
```

$$eqn := \{-l1 + x_4\, l3 + H4\, m1 + 2\, fx_4 = 0,\ -l2 - y_1\, l3 + 2\, fy1 = 0,\ -l2 - y_2\, l3 + 2\, fy_2 = 0,\ -l2 - y_3\, l3 - m2 + 2\, fy_3 = 0,\ -l1 + x_2\, l3 + 2\, fx2 = 0,\ -l2 - y_4\, l3 - m1 + 2\, fy_4 = 0,\ -l1 + x_3\, l3 + H3\, m2 + 2\, fx_3 = 0,\ -l1 + x_1\, l3 + 2\, fx1 = 0\}$$

Below we assign values to some of the motor forces assuming that the motors 3 and 4 are on maximum capability $fy_4 = K4(fx_4)$ and $fy_3 = K3(fx_3)$.

```
>   fy[4]:=K4;
```

$$fy_4 := K4$$

```
>   fy[3]:=K3;
```

$$fy_3 := K3$$

Recall that we solved for fx_1, fy_1, fx_2 in the first three equations of section 4.1.

```
>   fx1:=Fx-sum(fx[i],i=3..4)-fx2;
>   fy1:=Fy-sum(fy[i],i=2..4);
```

$$fx1 := Fx - fx_3 - fx_4 - fx2$$

$$fy1 := Fy - fy_2 - K3 - K4$$

```
>   eq:=Mz-sum(fx[i]*y[i]-fy[i]*x[i],i=3..4)+
>   fy[2]*x[2]+fy1*x[1]-fx1*y[1]-fx2*y[2];
>   fx2:=solve(eq,fx2);
```

$$eq := Mz - fx_3\, y_3 + K3\, x_3 - fx_4\, y_4 + K4\, x_4 + fy_2\, x_2 + (Fy - fy_2 - K3 - K4)\, x_1 - (Fx - fx_3 - fx_4 - fx2)\, y_1 - fx2\, y_2$$

$$fx2 := (Mz - fx_3\, y_3 + K3\, x_3 - fx_4\, y_4 + K4\, x_4 + fy_2\, x_2 + x_1\, Fy - x_1\, fy_2 - x_1\, K3 - x_1\, K4 - y_1\, Fx + y_1\, fx_3 + y_1\, fx_4) \Big/ (-y_1 + y_2)$$

```
>   Mz:=1:Fx:=2:Fy:=1:x[1]:=0:x[2]:=2:x[3]:=2:
x[4]:=2:y[1]:=1.2:y[2]:=1:y[3]:=2:y[4]:=2:
```

The solution to the Kuhn Tucker equations is

```
>   sol:=solve(eqn,{l1,l2,l3,m1,m2,fx[3],fx[4],
fy[2]});
```

$$sol := \{ml = -2.\,\frac{-248. + 230.\,K3 + 330.\,K4 + 81.\,H3\,K3 - 81.\,H3\,K4}{\%1},$$

$$l2 = -2.(-192. + 170.\,K3 + 170.\,K4 - 90.\,H3 + 189.\,H3\,K3 + 90.\,H3\,K4 - 90.\,H4 + 90.\,H4\,K3 + 189.\,H4\,K4)/(\%1),$$

$$m2 = 2.\,\frac{248. - 330.\,K3 - 230.\,K4 - 81.\,H4\,K4 + 81.\,H4\,K3}{\%1},$$

$$fx_4 = (180.\,H4\,K4 + 81.\,H4\,H3\,K3 + 60. + 150.\,H4\,K3 - 80.\,H3\,K4 + 125.\,H3 - 123.\,H4 - 100.\,K3 - 150.\,H3\,K3 - 100.\,K4 - 81.\,H4\,H3\,K4)/(\%1),$$

$$fx_3 = -1.(150.\,H4\,K4 + 81.\,H4\,H3\,K3 - 60. + 80.\,H4\,K3 - 150.\,H3\,K4 + 123.\,H3 - 125.\,H4 - 180.\,H3\,K3 - 81.\,H4\,H3\,K4 + 100.\,K3 + 100.\,K4)/(\%1),$$

$$fy_2 = -1.(28. - 30.\,K3 - 30.\,K4 - 45.\,H3 + 54.\,H3\,K3 + 45.\,H3\,K4 - 45.\,H4 + 45.\,H4\,K3 + 54.\,H4\,K4)/(\%1),$$

$$l3 = 10.(-44. + 40.\,K3 + 40.\,K4 - 9.\,H3 + 27.\,H3\,K3 + 9.\,H3\,K4 - 9.\,H4 + 9.\,H4\,K3 + 27.\,H4\,K4)/(\%1),$$

$$ll = 10.(-76. + 60.\,K3 + 60.\,K4 + 7.\,H3 + 24.\,H3\,K3 + 2.\,H3\,K4 + 7.\,H4 + 2.\,H4\,K3 + 24.\,H4\,K4)/(\%1)\}$$

$$\%1 := -100. + 81.\,H3 + 81.\,H4$$

The solution to the optimization problem is then obtained by solving the following two equations

```
>   equa1:=fx[4]=subs(sol,fx[4]);
```

$$equal := fx_4 = (180.\,H4\,K4 + 81.\,H4\,H3\,K3 + 60. + 150.\,H4\,K3 - 80.\,H3\,K4 + 125.\,H3 - 123.\,H4 - 100.\,K3 - 150.\,H3\,K3 - 100.\,K4 - 81.\,H4\,H3\,K4)/(-100. + 81.\,H3 + 81.\,H4)$$

```
>   equa2fx[3]=subs(sol,fx[3]);
```

$$equa2 := fx_3 = -1.(150.\,H4\,K4 + 81.\,H4\,H3\,K3 - 60. + 80.\,H4\,K3 - 150.\,H3\,K4 + 123.\,H3 - 125.\,H4 - 180.\,H3\,K3 - 81.\,H4\,H3\,K4 + 100.\,K3 + 100.\,K4)/(-100. + 81.\,H3 + 81.\,H4)$$

for fx_3 and fx_4. So we have to insert the expressions for Hi and Ki. When the two equations above are solved for fx_3, fx_4 by for instance the Newton Raphson method we can substitute the values of the remaining variables from the solution formulas in *sol*.

5 Conclusions

We have shown in this report that the holding problem for a given configuration of thrusters on a ship can be solved in a time that increases at most linearly with the number of thrusters n. This improves on the previous method of solution which, it seems, required a number of calculations that increases exponentially with n.

The holding capability has some peculiar features. Here we point out only one of these. By comparing cases I – III, we see that the solution for $(\theta_1, \ldots, \theta_n)$ changes discontinuously when one thruster goes from being on full power to slightly below full power. Consistent with this observation, the numerical results in Section 4 show a similar discontinuous change in the optimal position of the thrusters (although it is a slightly different Lagrangian which is considered).

We have made no restriction on the force (F_x, F_y) and the torque M_z. Once this general problem has been solved, it is straightforward to solve a special case where the restrictions are present. Such restrictions arise in practice because, for instance, the wind will result in both a force on the centre of mass and a torque around this point. The exact form of the relationship between force and torque depends on the ship considered. However, to see how this special case can be solved in principle, suppose that the torque is a function of the force,

$$M_z = f(F_x, F_y). \tag{15}$$

This equation describes a two-dimensional surface in the three-dimensional space (F_x, F_y, M_z). Recall that the solution to the holding problem in the general case provided us with another two-dimensional surface. The solution to the holding problem in the special case where a relation of the type in Eq. (15), holds is the intersection of the two surfaces. (By the way, it can be argued that the intersection is non-empty. Because, on physical grounds, f is continuous, $f(0,0) = 0$ and $f(x, y)$ must be unbounded, in the general case, in (x, y)).

Bibliography

[1] Chong, E.K.P. & Zak, S.H., *An introduction to optimization*, Wiley-Interscience, 1995.

[2] Courant, R. & Hilbert, D., *Methods of mathematical physics. Vol. I.* Interscience Publishers, Inc., New York, N.Y., 1953.

[3] Larsen, J. C., `http://www.matfys.kvl.dk/biomodel/notes`

[4] Petersen, H.G. Supplerende noter - AM02 IMADA Odense University, Denmark, 1995. (Lecture notes on optimization in Danish).

[5] Whittle, P., *Optimization under constraints. Theory and applications of nonlinear programming.*, Wiley-Interscience, London-New York-Sydney, 1971.

The Group Working on the Problem

Jens Christian Larsen, [jlarsen@dina.kvl.dk]

Henrik Rasmussen, [rasmusse@maths.ox.ac.uk]

Peter Sinding, [psi@danmar.dk]

Ove Skovgaard, [Ove.Skovgaard@mat.dtu.dk]

Allan Sørensen, [allans@mip.ou.dk]

Grundfos: Chlorination of Swimming Pools

Viggo Andreasen, Morten Brøns, Poul Hjorth, John Hogan, and David Wood

1 Introduction

Grundfos asked for a model, describing the problem of mixing chemicals, being dosed into water systems, to be developed. The application of the model should be dedicated to dosing aqueous solution of chlorine into swimming pools.

The first thing to be decided is the type of model we are looking for; in particular whether this is a diffusion dominated problem (where spatial gradients and diffusion times are important) or a sink-source type of ordinary differential equation problem like the CSTR (Continuously Stirred Tank Reaktor).

Swimming pools as a rule are designed specifically to mix the re-circulated water well, i.e., much faster than the typical diffusion time associated with the pool size. Such a rapid mixing is accomplished by having the chlorinated water re-enter the pool through a large (30–80) number of jets, evenly spaced across the pool bottom. The actual mixing time in a particular public pool is monitored as a part of the routine whenever the pool water is to be drained prior to a filling with fresh water.

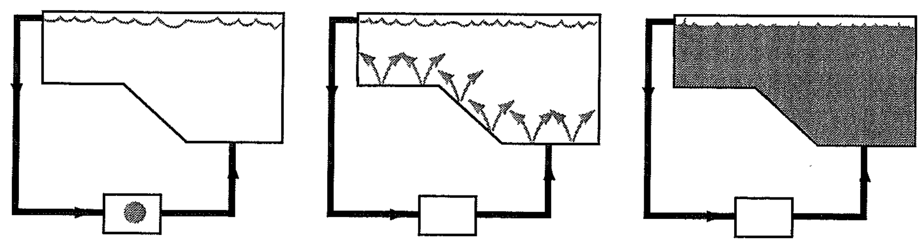


Figure 1: Mixing jets at bottom of pool.

As the final operation on the 'dirty' water, a single blob of dye is injected into the water entry system, and the subsequent spreading of the dye throughout the pool is observed and timed, see figure 1. A well functioning jet system (such as e.g., in the *Lyngby Svømmehal*) mixes the dye evenly in the entire pool in 2–3 minutes; a