Optimization and Algorithms

Aircraft Collision Avoidance

Andreia Chagas Catarina Gaspar Inês Lourenço
andreiachagas@msn.com catarina.cgaspar@gmail.com
ineslourenco@tecnico.ulisboa.pt
Instituto Superior Técnico

***Abstract*—Our goal in this paper is to solve an optimization problem regarding the trajectories of multiple aircraft avoiding collisions. This project is divided in two parts. The first one concerns the avoidance of fixed obstacles that may be intercepting the trajectory of an aircraft. On a second phase, it is desired to avoid other aircraft (moving obstacles) instead of fixed obstacles. In both phases, it is computed a first trajectory without concerning any restrictions about avoiding collision. Then, multiple trajectories are iteratively estimated until all constraints are satisfied, obtaining the final trajectory. Based on a previous work, a new algorithm was formulated in order to solve the problem in CVX software in Matlab. CVX is a Matlab-based modeling system for convex optimization. CVX allows constraints and objectives to be specified using standard Matlab expression syntax.**

I. Introduction

The scope of this project, path-planning for vehicles, is quite in use nowadays in a wide range of applications. One example that is currently being developed are autonomous cars that have to be able to avoid obstacles in their path. These cars must be able to identify obstacles and calculate the best trajectory between the starting and finishing points, keeping a safety distance from it, saving energy or the fuel consumption, and ensuring the comfort of its passengers. Air traffic management is another application.

In the original formulation the optimization problem consists on having several aircraft moving at the same time, and the goal is to make them reach its destination in the less possible time interval. This implies some constraints regarding the turning rate, the aircraft's maximum speed and maximum applied force. In order to formulate a linearly solvable problem, all the aircraft are considered as a point of mass, and then some conditions on the behaviour of the system are imposed. The first condition concerns its environment, and this problem appears in a context in which one or more aircraft are moving in a planar surface, which corresponds in practice to an adequate approach to real life problems that is a flight with constant altitude. Secondly, there will be a maximum speed limit to its movement, and also a limited magnitude force acting on the aircraft, which introduces a constraint in the turning rate (w) since it is not possible to do a very thigh turn instantaneously.

In order for the aircraft to avoid obstacles, there must be an area around it that can not be intercepted by any object. This region can be approximated by linear constraints, but is represented by a set of binary variables (c_i), that become decision variables for the optimization. A binary variable (b_i) is also used to know weather or not the final destination has already been reached in some iteration of the algorithm, before the maximum time was reached, but does not require that the vehicles finish at the same time.

In the end, the problem consists on minimizing the time elapsed at each iteration, as well as the force applied to each one, ensuring that there only is one unique solution to the problem. The optimization variables are s,u,b,c, subject to the constraints already mentioned.

However, the resolution of this kind of problem is quite hard since the optimization is non-convex, and this brings a problem since CVX does not have the capacity to deal with binary variables. For this reason, there was a reformulation of the problem, aiming to make it a convex optimization problem.

Instead of a single optimization problem, there will be an algorithm solving the optimization problems in some of its iterations. It consists on the minimization of both the distance travelled by the aircraft from a starting point to a destination point, and the total of the applied forces at each moment.

In this paper the new formulation will be studied, and it is divided in two parts. In the first one, an algorithm was developed in order to make one aircraft travel from one initial point to a specified destination, avoiding the collision with a fixed obstacle. Whereas on the second phase the algorithm has to plan the trajectories of several aircraft, ensuring that they never get too close to each other and therefore do not collide in any point of the trajectory.

II. Problem formulation

The same way as in the original problem, in the new one the problem is still approached in two dimensions, and it remains necessary to consider the aircraft a point of mass in which some constraints are applied. Although it is desired to minimize the travelling distance instead of the time, it is still favourable to be at the maximum speed for as long as possible. This implies that there has to be defined a maximum value of speed (V_{max}) and force (f_{max}), in both x and y directions, and it is important to get the ideal combination of the variables in order to obtain the fastest turn possible, which will lead to a faster trajectory. The expression used to minimize the distance between two points is given by

$$\underset{p,v,f}{minimize} \quad \alpha_1 \left\|p - p_{final}\right\|_1 + \epsilon(\left\|f_x\right\|_1 + \left\|f_y\right\|_1) \qquad (1)$$

where p_{final} are the coordinates (x_{final}, y_{final}) where the aircraft should be after the end of the optimization, and (f_x,

f_y) is a vector of the components of the applied forces. ~~This means that in every iteration the algorithm will try to minimize both the distance to final point, and the applied forces.~~ The constraints include the initial conditions (initial position ($p(1) = p_{initial}$), initial speed ($v(1) = v_{initial}$), and initial force ($f(1) = f_{initial}$)), the restriction of the maximum speed and force, and also two expressions that indicate the position and velocity of the aircraft in the next iteration of the algorithm (p_{i+1} and v_{i+1}), that are given by

$$\begin{aligned} p(i+1) &= p(i) + v(i) \\ v(i+1) &= v(i) + f(i) \end{aligned} \tag{2}$$

Where the next position (p_{i+1}) corresponds to the previous one (p_i) plus the velocity in the previous one (v_i) multiplied by the time interval, that here is the difference between two consecutive iterations ($i+1-i = 1$). The same logic is applied to the velocity. In the minimization expressions every term is multiplied by constant factors (α, β, ϵ), that are responsible for assigning different weights to each of the terms of the optimization.

When obstacles are intercepting the aircraft's trajectory, in order to avoid them, there must be an area around the aircraft that can not be intercepted by any object and a consequent adjustment of that trajectory is needed. Unlike the original formulation, in this implementation it is done adding a convex constraint that ensures that the security distance of the obstacles is not transposed, instead of using a binary variable. This implies adding a term to the minimization function that saves the new points of the new trajectory (p_{new}),

$$\begin{array}{ll} \underset{p_{new}, v_{new}, f_{new}}{minimize} & \alpha_2 \left\| p_{new} - p_{final} \right\|_1 + \beta \left\| p_{new} - p \right\|_1 + \\ & +\epsilon(\left\| f_{new_x} \right\|_1 + \left\| f_{new_y} \right\|_1) \\ \text{subject to} & p_{new}(1) = (x_i y_i) \\ & v_{new}(1) = (v_{x_i} v_{yi}) \\ & f_{new}(1) = (f_{x_i} f_{yi}) \\ & p_{new}(i+1) = p_{new}(i) + v_{new}(i) \\ & v_{new}(i+1) = v_{new}(i) + f_{new}(i) \\ & \left\| f_{new_x} \right\| \leq f_{max} \\ & \left\| f_{new_y} \right\| \leq f_{max} \\ & \left\| v_{new_x} \right\| \leq v_{max} \\ & \left\| v_{new_y} \right\| \leq v_{max} \\ & \frac{(p(I)-o)}{\|p(I)-o\|}^{\top} (p_{new}(I) - o) \geq d \end{array} \tag{3}$$

where I is a variable that saves the instance of the first violation of the security distance in each iteration, and has the value 0 when no object intercepts the trajectory, and d is the safety distance.

Apart from this, it can be considered that the obstacles can have its own trajectories, which corresponds to the case of multiple aircraft travelling from one position to another, without colliding with each other. For each aircraft there is a pair of coordinates (x, y) describing its position, (v_x, v_y) describing its velocity, and (f_x, f_y) describing the applied forces. This is done changing the position of the object for the position of the other aircraft in the last constraint.

III. Approach

In order to solve the first optimization problem stated previously in equation X, firstly the obstacles are neglected, and then an optimal trajectory that minimizes the distance between the starting and the finishing points is found, as well as the total force applied to the aircraft. Then, the object is introduced and there is the need to find out if, at any point, the previous estimated trajectory collides with the fixed object. In other words, if, at any time, the safety distance is violated. If not, then the optimal trajectory has been found and the problem is solved (the ~~optimization~~ algorithm ends). If there is violation, then the trajectory has to be rectified in order to avoid collision. In this case, it was defined that the avoidance of the collision would be by means of deviating to a safety zone the point of the trajectory that first violates the safety distance. This means that the new estimated trajectory should be as close to the previous one as possible, and it also should take into account, as before, that the distance to its destination and the sum of the applied forces should be minimized. This way the problem to minimize should be the one stated in equation 3 subject to the indicated constraints, where one of them guarantees that the safety distance in the first violating instant is fulfilled, and is given by:

$$\frac{(p(I)-o)}{\|p(I)-o\|}^{\top} (p_{new}(I) - o) \geq d \tag{4}$$

where p_{new} represents the new rectified trajectory, p represents the previous trajectory, o represents the obstacle's position, d is the radius of the obstacle's safety zone, and I is the time instant when the previous trajectory first violates the safety distance. Geometrically, the expression (4) states that the internal product (scalar product) between two vectors should be higher or equal to the safety distance. The first vector is a normalized vector defined by the position of the aircraft at time I during the previous trajectory, and the obstacle's position. The second vector (new vector) is defined by the correspondent new position of the aircraft (during the rectified trajectory) and the obstacle's position. Thus, this internal (scalar) product matches the projection of the new vector along the direction of the normalized vector, so it corresponds to the norm of the vector associated to the new position. This norm is the distance between the new point and the obstacle and it has to be ensured that it is greater or equal to the safety distance. This rectification is repeated until the safety distance is satisfied for each point of the trajectory.

As for the second optimization problem, in which it is required the avoidance of the collision of several aircraft, the procedure for optimization is similar to the procedure for the first problem. Firstly each aircraft's trajectory is computed independently, then the first aircraft's trajectory is fixed and it is checked whether the second collides with the first one or not. If it does, then the second aircraft's trajectory needs rectification(s). This rectification is like the one on the first problem, but instead of having a fixed obstacle we now have a moving one: the other aircraft. This means that in equation (4),

o is replaced by the other aircraft's trajectory point (the point where the two aircraft in question collide with each other):

$$\frac{(p_j(I) - p_k(I))}{\|p_j(I) - p_k(I)\|}^{\top} (p_{j_{new}}(I) - p_k(I))) \geq d \qquad (5)$$

where j is the aircraft whose trajectory is being rectified and k is the aircraft that is being considered as the moving obstacle. In this situation $j = 2$ and $k = 1$. Once this second trajectory is rectified, it is also fixed, and the algorithm proceeds to rectifying the next aircraft's trajectory. When there are more than two aircraft, it is important to notice that when correcting the trajectory of aircraft number N, it must be done concerning the trajectory of all aircraft number $\{1, 2, ..., N-1\}$. For example, when rectifying the third trajectory, there is the need to check if there is any collision between the third and both the first one and the second one. This optimization problem is solved when all aircraft's trajectories computed have, at every point, fulfilled the aircraft's safety distance.

IV. Numerical results

The ~~minimization~~ variables are: the position taken by the aircraft at each moment, the velocity and the applied force, all with both its x and y components. For the first optimization problem, in which only one aircraft is considered, there are 3 minimization variables. As the number of aircraft increases (second optimization problem), so does the number of minimization variables and therefore for n aircraft there are $n \times 3$ ~~minimization~~ variables.

In all the examples shown in this report each aircraft has to reach its destination in a maximum of 100 time instants ($T = 100$), the maximum velocity and applied force are respectively 0.8 and 0.3. Also, the initial velocity is 0.3 for the x direction and 0 for the y. And the initial force is 0 for both x and y directions. The axis of all the images show the coordinates x and y, and the position of the aircraft in each time instant is represented by a small circle, being its trajectory the sequence of all the small circles.

In the first part of the project there is only one aircraft moving and one fixed object in its trajectory, which represents the collision problem shown in Figure 1, that is computed by the equation (1). α_1 has the value 1 and ϵ is 0.001. This way, in blue, is shown the trajectory of the single aircraft, travelling from the initial position (0,2) to the final position (10,6) and in orange is the boundary of the security region of the obstacle, which is located at (6,3), that must be avoided.

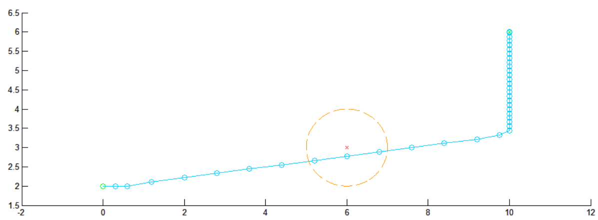

Fig. 1. First trajectory simulated

As explained on section III. Approach, the algorithm simulates the trajectory and saves the instant that first violates the safety distance to the object. In this case, it happened at the instant $I = 9$. Based on this knowledge, a new trajectory will be calculated in several iterations (pink trajectories), until it finally surrounds the safety area, in the green and final trajectory as it can be seen in Figure 2.

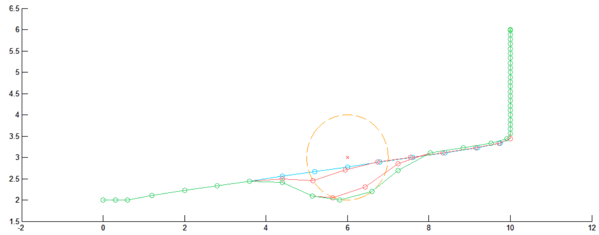

Fig. 2. All trajectories before the final one and the final one (green line) (E=0.001)

In the previous example (Figure 1 and 2) ϵ, which is the weight of the component of the force, had the value 0.001, and α_2 was 0, which means that the algorithm, when rectifying trajectories, never takes into account that the total distance should be minimized, it only cares about the similarity with the original trajectory and the minimization of the applied forces. In Figure 3 all variables kept the same values as before, except the ϵ, that is decreased to 10^{-8}. It can be concluded that, as expected, the minimization of the total travelled distance is now privileged when compared to the minimization of the applied forces, which results in a faster speed, and a shortest path, as can be verified in Figure 3. Before, if the aircraft was travelling at a low speed, the price to accelerate was too high and therefore the aircraft would remain at a low speed, avoiding accelerations, even if that implied that the trajectory would be longer than it could be.

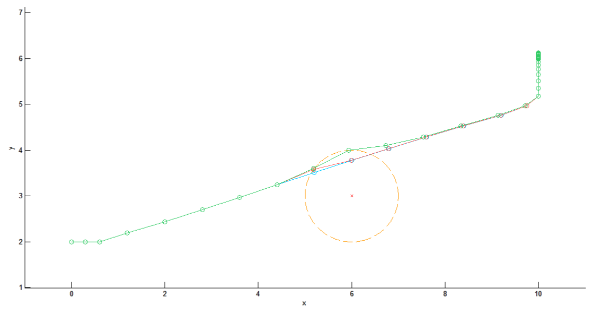

Fig. 3. All trajectories before the final one and the final one (green line) (E=10e-8)

In the second part of the program the algorithm will deal with multiple moving aircraft. The goal is to calculate the optimal trajectory between them, ensuring they keep a safety distance with one unit radius between them.

Figure 4 shows the trajectories of three aircraft. The first one starts at point (0,2) and ends at point (10,5) (blue line), the second one leaves from point (11,2.5) and ends at point (0,2.5) (yellow line) and the last one starts at point (2,3.5) and ends at point (11,3.5) (green line). The continuous line shows the initial calculated ideal trajectories for each aircraft, and the dashed ones are the improvements along the iterations.

The trajectory of aircraft number one (blue trajectory) is fixed right on the beginning of the optimization as explained in section III. Approach, and then the second one (yellow trajectory) is corrected only based on the collisions with the first one. Once the yellow trajectory is also fixed, the third one (green trajectory) is corrected looking at both the first one and the second one.

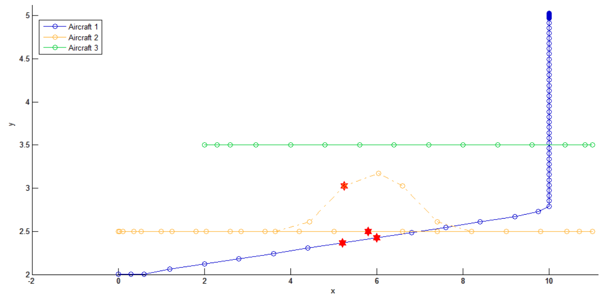

Fig. 4. First correction for the trajectory.

Figure (4) represents the first correction of the yellow trajectory (correspondent to aircraft number two), because this aircraft violates the safety distance of the aircraft number one, this occurs at the 9^{th} instant. This is represented by the two red points in both the trajectories at that instant. The other two red points appear because even after correcting the first time the yellow trajectory, there is still another collision with the blue one at instant 10. In Figure 5 is shown the whole procedure of the algorithm, being that when any point of one trajectory collide with the others, the finals trajectories are drawn in bold (Figure 5).

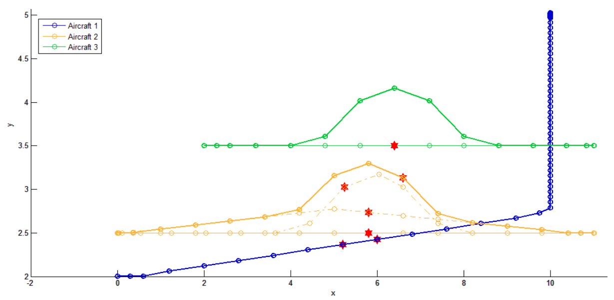

Fig. 5. Positions of multiple aircraft, in different instants.

As a matter of example, in Figure 6 is a test of the algorithm to a different set of initial and final positions, regarding two aircraft only. Both α and β are now equal to one, which means that both the minimization of the total distance and the minimization of the divergence of the corrected trajectory to the original one have the same importance.

Aircraft 1, blue line, starts at point (0,2) and ends at point (10,5), the other one, orange line, leaves from point (4,4) and ends at point (8,2).

Once again the first trajectory is fixed and the second is iteratively adjusted. It can be seen that the optimized yellow trajectory makes the aircraft spend time looping in order to

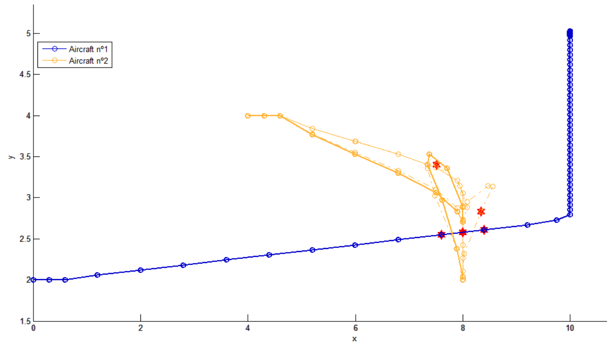

Fig. 6. All trajectories before the final one (green line).

allow the other aircraft to move in its way. It can be concluded that this optimization is not optimal given that the aircraft is walking back, instead of deviating right-a-way to a more favourable and secure point.

V. Conclusions

- Desviam-se bem da maioria dos obstaculos, e avies tb

- Mas qd temos muito compicados o programa no optimo e falha

- O alfa no faz diferena ao desviar de um obstculo. Ou seja, termos em conta ou no a distancia minima na correco da trajectoria no faz diferena.

- ao dar mais importncia ao parametro da proximidade a trajectoria anterior, verifica-se que a nova trajectoria bastante identica anterior e por vezes no se obtem a soluo optima pois no ser a trajectoria mais rapida.

- H coisas que so mesmo do programa e no conseguimos resolver, -nos dada uma soluo sub-optima.

- Fazendo vrias experiłncias a alterando valores das componentes ao dar mais importncia ao parmetro da minimizao da distncia, α, aircraft desloca-se mais rapidamente

- Suboptimal optimization. there will be errors that we can not control and sometimes the behaviour of the system is different from the expected

References

[1] Tobias Oetiker, *The (not sure) short introduction to LaTeX*. available at http://tobi.oetiker.ch/lshort/lshort.pdf 2011