Optimal location of dynamic military areas within civil aviation traffic

Ning Wang, Daniel Delahaye, Marcel Mongeau, Alexandre Gondran

[EN-A-044] Optimal location of dynamic military areas within civil aviation traffic

N. Wang, D. Delahaye, M. Mongeau, A. Gondran

ENAC, Université de Toulouse, France
[ningwang | delahaye | mongeau | gondran]@recherche.enac.fr

Abstract: In this study, we focus on the problem of locating optimally dynamic military areas with the aim of minimizing the number of civil flight trajectories potentially impacted by the military activity, and the distance between the military area and the military base. We model the military areas by 2D geometry shapes with a vertical extension associated to given flight levels during the temporary area-activation time window. We propose a mathematical formulation of this problem as a constrained-optimization problem. We then introduce a global-optimization methodology based on a simulated annealing algorithm featuring tailored neighborhood-search strategies and an astute computational evaluation of the otherwise costly objective function. This is applied to one day of French traffic involving 8,836 civil flights. The results show that the proposed method is efficient to locate the military area that is nearest from the military base, while minimizing the potential impact on civil flight trajectories.

Keywords: dynamic military area, optimization, location problem, simulated annealing algorithm

1 Introduction

Figure 1: Potential military areas in lower French airspace

Along with the fast development of aviation technology, air traffic experienced its highest growth over the last five years [1]. Traditionally, prohibited airspace or no-fly areas, where no civil flights are allowed at any time, are established for security reasons or military activities. Fig. 1 shows the real potential military areas (represented by the pink and orange polygons) in the lower French airspace. If we zoom in around the Charles de Gaulle airport (Paris), seven potential military areas are found, represented in Fig. 2 by yellow polygons of various shapes surrounded by fences, and whose dimensions range from 40 NM to 120 NM. These areas are placed so as to avoid the main civilian traffic flow (represented by different colors of lines). However potential conflicts still exists for a large number of trajectories. Therefore, military areas have become significant constraints for civilian traffic.

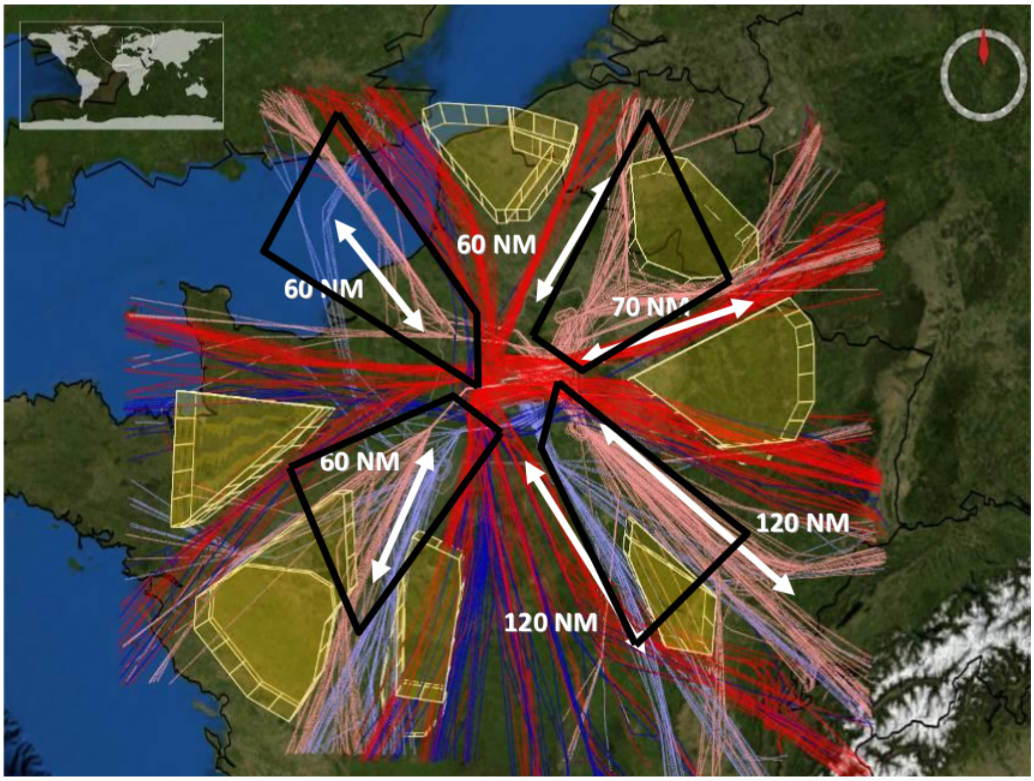

Figure 2: Military areas (yellow polygons surrounded by fences) around Charles de Gaulle airport, Paris

To meet the increasing civilian traffic demand and to improve airspace utilization, implementing temporary/dynamic prohibited areas to perform military training or missions for short periods of time seems to be a better choice. A civil-military coordination program is thereby necessary for airspace planners to take the military activities into consideration. In a static airspace planning process, the military part provides the time and location information of the upcoming military missions online one or several days before the operation. Airspace planners take these constraints into account to come up with an operational airspace use plan and then publish it. Then, any changes after publication such as weather changes, cancellations of missions etc. might not be captured, leading thereby to a waste of resources. To realize flexible airspace management, a dynamic airspace planning process is preferred: the coordination process continues after the airspace use plan is published, until a time much closer to the time of operations. This yields increased benefit and a better use of resources through more accurate data while maintaining full flexibility for military operations. In this case, the airspace management is performed at the tactical level.

Ideally, the airspace planning platform is online and in real time. It should allow civil and military parts for airspace bookings and coordinations. It can also carry out analyzing functionalities such as conflict detection, automation of tasks such as NOTAM (Notice to Airmen) requests for pilots, airspace use plan drafting, and visualization display for airspace planners. With such an airspace planning platform, common situation awareness are available for tactical civil-military coordination. Real-time activation and de-activation of airspace are viable based on planning and acknowledgements of the air traffic controllers in charge. Finally, airspace status can be displayed on different interfaces of the ATM system.

The left part of Fig. 3 displays a scenario in which civilian traffic flow (represented by white two-way arrows) overlaps a military area (represented by the blue polygon). To tackle this issue, various methods are proposed: one can divide the current military area into subsets, consider for example the three parts A, B and C, as presented by Fig. 3 left. One can then active only the A, C parts during the time (e.g. at night) during which civilian traffic is not active. One can then relocate the military area by airspace discretization, and find the grids of airspace that civilian traffic does not occupy (Fig. 3 center). One can also relocate the military area dynamically in time and space, taking the civilian traffic into consideration (Fig. 3 right). The first two methods are relatively easy to implement, while having recourse to the Dynamic Military Areas (DMA) constitutes a scientific challenge. In this study, we carry out a preliminary study on such a DMA approach. The objective of this study is to allow optimal military airspace reservations in time, flight levels and/or geographical location without limiting mission effectiveness while enabling a more efficient use of the airspace for civilian flights.

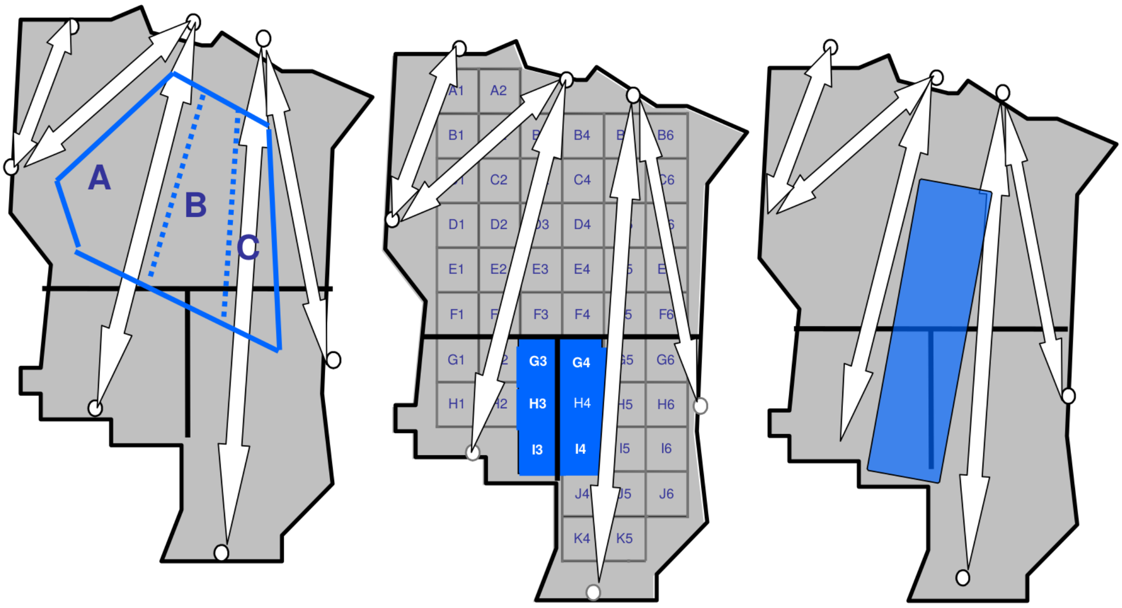

Figure 3: Variable and dynamic military areas

In this article, we concentrate on DMA for military missions that relies on designing an airspace volume that envelopes the whole military mission trajectories. It features user-defined lateral and vertical dimensions, along with user-defined activation-time duration. Our decision variables include the 3D geographical location of the the military area related to the military base position. The activation time is also an optimization variable, it can be chosen within a user-defined time window. These parameters will be computed so as to minimize both the amount of civilian flights overlapping the military area (at the strategic level) and the distance between the DMA and the military base.

To our knowledge, there is no previous published study on the military area location problem. Nevertheless, one is tempted to consider our problem as a special case of the single facility location problem. For this type of problem, “quick and dirty” methods are recommended [2]. For our problem, approximate models are used due to limited input data available for decision makers. Contrary to the classical facility problem, here the facility is considered “toxic”, therefore one aims at minimizing the harmful effect (the impact on civil trajectories in our problem) inside the facility (the military area) coverage range. Another specific feature of our problem is the fact that it deals with *continuous* candidate locations. The first survey on continuous location problems was conducted by F. Plastria. In his work [3], exhaustive topics on continuous location theory are discussed. A recent survey on continuous location-allocation problems was performed by J. Brimberg [4] in 2008; his work confirms that heuristic methods contributed significantly to improving the methods addressing this type of problem.

The remaining of this article is organized as follows: in Section 2, a mathematical model for the military area optimal location problem is proposed. Section 3 introduces a simulated annealing algorithm to address this problem with dedicated neighborhood operators. Several test results on the French airspace are presented, and performance of the algorithm is analyzed in Section 4. Finally, conclusions are drawn along with future perspectives for military area location problems.

2 Mathematical model

In France, there are more than 8,000 civil flights in one day. Their trajectories cover almost the whole airspace above the French territory, as showed in Fig. 4. An example of a location of Military Area (noted as MA in the following sections) is represented by the green rectangle in this figure, and the position of the military base is the center of the blue circle. Planned flights passing through this rectangle during the activation time of the MA are affected and have to be modified. The MA is to be relocated inside this circle at different time windows so as to minimize the impacts on the civil flights planned during this same period. Moreover, from the fuel consumption saving point of view, minimizing the distance between the MA and its military base is also desirable.

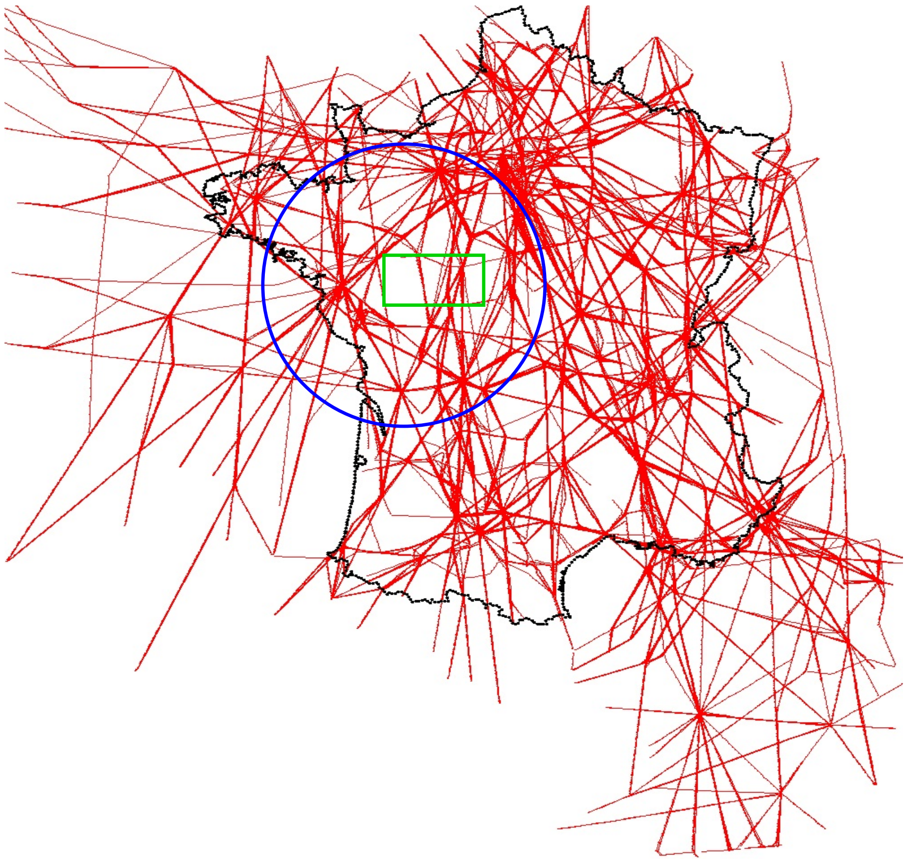

Figure 4: Example of an MA location (green rectangle) near the military base (center of blue circle) within civil flight trajectories (red lines) over France

In this section, we model the MA by a 2D-geometry shape (a polygon) with a cylindrical vertical extension (i.e. a 3D cylinder with a polygonal section), and then we formulate our MA location problem as a constrained optimization problem. Remark that although one should properly speak of a military *volume*, we shall stick in the sequel to the more familiar MA terminology.

In this study, we aim to locate an MA for one given military base; the position of this military base is given as input data. To simplify the exposition, the shape of this MA is defined as a rectangle with a given vertical extension, i.e. a cuboid with given length L, width W and height H. It has also a given activation-time duration D. We also assume, as showed in Fig. 5, that it can then be rotated and/or moved around the military base, where (ρ, θ) are the polar coordinates of this MA (center of the rectangle), and ϕ is the rotation angle of the MA. We also denote the altitude of the lowest point of this MA as z, considering the military base as the origin of the space, and its starting activation time as t. To summarize, the decision variables are ρ, θ, ϕ, z and t. In theory, all five variables are continuous variables. In practice, aircraft follow discretized Flight Levels (FL) in order to ensure safe vertical separation. Here, we therefore consider discrete elementary altitude shifts $\delta z = 5$ FL (1FL = 100 feet), and we consider the discrete set $Z := \{z_{min} + \delta z, z_{min} + 2\delta z, \ldots, z_{min} + J\delta z\}$ for the possible values of z, where $J = \lfloor (z_{max} - z_{min})/\delta z \rfloor$ ($\lfloor z \rfloor$ denotes the largest integer greater than or equal to z), z_{min} and z_{max} are respectively the lower and upper bounds of altitude constraints for the MA. To simplify the time allocation, we also consider discrete time shifts $\delta t = 5$ minutes, and we define the discrete set $T := \{t_{min} + \delta t, t_{min} + 2\delta t, \ldots, t_{min} + K\delta z\}$ for the possible values of t, where $K = (\lfloor t_{max} - t_{min})/\delta t \rfloor$, t_{min} and t_{max} are respectively the lower and upper bounds of activation time constraints for the MA. Let us denote ρ_{max} to be the maximum allowed distance between the center of the MA and the center of the military base. Let MA(x) denote the 4D (time + space) MA defined by an instantiation of our five-dimensional vector of decision variables $x = (\rho, \theta, \phi, z, t)$.

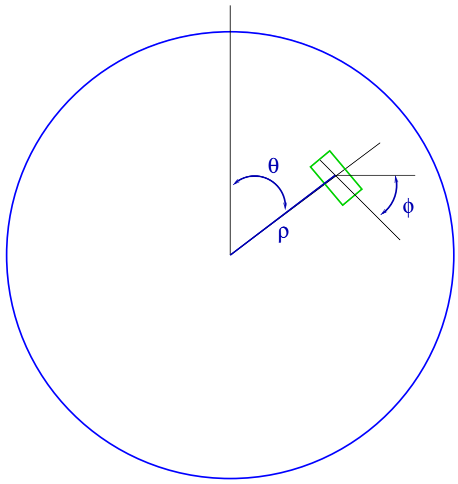

Figure 5: MA location model: (ρ, θ) are the polar coordinates of the center of the MA (green rectangle), ϕ is the rotation angle of the MA

The civil flight (4D) trajectories are given under the form of discretized sample points $P_i \in \mathbb{R}^4, i = 1, 2, \ldots, M$, at intervals of 15 seconds, where M is the total (cumulated) number of sample points for all the trajectories. In order to minimize the impact of the military activity on the civil traffic, we aim at positioning the MA (in 4D) so as to minimize the number of trajectory sample points intersecting it. We also wish to minimize the distance between the MA and the military base. This is therefore a bi-objective optimization problem that we propose to model as:

$$\begin{aligned}
\underset{x=(\rho,\theta,\phi,z,t)}{\text{minimize}} \quad & f(x) = \sum_{i=1}^{M} 1_{P_i \in \mathrm{MA}(x)} + \mu.\rho && (1)\\
\text{subject to} \quad & 0 \le \rho \le \rho_{max} && (2)\\
& 0 \le \theta \le 2\pi && (3)\\
& 0 \le \phi \le 2\pi && (4)\\
& z \in Z && (5)\\
& t \in T && (6)
\end{aligned}$$

where the indicator function $1_{P_i \in \mathrm{MA}(x)}$ is equal to 1 if $P_i \in \mathrm{MA}(x)$ and is equal to zero otherwise. Hence $\sum_{i=1}^{M} 1_{P_i \in \mathrm{MA}(x)}$ corresponds to the number of 4D sample trajectory points that lie inside the MA, and μ is a weighting parameter whose value is to be set by the decider according to his priority between the two criteria: distance ρ (expressed in NM) vs. traffic impact. In our tests, we use $\mu = 1$ so as to prioritize the impact on civil traffic. Remark that this model could easily be adapted so as to evaluate the traffic impact by counting rather the number of *trajectories* intersecting the MA.

When we consider only minimizing traffic impact (case where the user sets $\mu = 0$), this problem can also be considered as the dual problem of the classical maximum coverage problem involving only one set (one MA in our case). This one-set situation removes the combinatorial nature of the maximum coverage problem. Nevertheless, instead of the typical discrete candidate sets or a planar (2-dimensional) continuous location problem, we deal here with a 5-dimensional location problem involving 3 continuous location parameters. The difficulty of this continuous optimization problem comes here from the fact that the objective function is non-differentiable and costly to evaluate (a black-box optimization problem), as counting the number of intersecting points is time consuming and this computation must be performed at every iteration. This is why we propose here a heuristic resolution method rather then having recourse to an exact approach.

3 Simulated annealing algorithm

Simulated Annealing (SA) is a typical Monte Carlo global optimization algorithm. It is a paradigm that simulates the relationship between the internal energy and the temperature T during the process of metal cooling [5]. Metal is in liquid state at high temperature; its internal energy is at its highest level but this state is highly unstable because of the high disorder of its atomic structure. If a fast cooling procedure is applied, then temper phenomena appears: internal energy quickly drops down but does not converge towards a globally minimal value; the metal has then an amorphous atomic structure with high disorder. However, during a slow cooling process, at each temperature, the internal energy has some probability to reach a locally minimal value. When the temperature reaches room temperature, the internal energy of the metal stabilizes at a minimal value; metal has then a crystal atomic structure.

At the beginning of the cooling process, an initial state x_0 must be provided by the user. This initial state can be randomly generated or can be particularly chosen to improve the SA performance; for instance, using prior knowledge on the problem, or using any heuristic. The temperature decrease at each iteration i is dealt with some *cooling schedule* such as: $T_i = \alpha^i T_0$, where T_i is the temperature at iteration i, T_0 is the initial temperature chosen by the user, and $\alpha \in [0, 1]$ is a user-defined cooling parameter. At each iteration/temperature, N_t number of *transitions* (from one solution to another) are performed. At each transition, a neighbor state x' of the current state x is generated by some neighborhood operator *getNeighbor*(x). This new state is then evaluated through the objective function f. If the new state is better than the previous state ($f(x') \leq f(x)$), it is accepted. When the new state is worse than the current state, it is accepted with some probability $I\!P(x, x', T)$ related to the current temperature and the objective-function degradation. The most-used acceptance probability for a minimization problem is:

$$I\!P(x, x', T) = \exp \frac{(f(x') - f(x))}{T}$$

The idea is that a bad transition is more likely to be accepted at high temperatures.

After acceptance or rejection of the new state, the algorithm proceeds to the next transition. The algorithm stops when T_i goes below some pre-defined final temperature T_f (for instance, we use $T_f = 0.0001 T_0$ in our tests), or some pre-defined target objective-function value is reached, or when after performing N iterations, where N is the maximum number of iterations decided by the user. SA is given in Algorithm 1:

Algorithm 1: Simulated annealing algorithm for a minimization problem

$T \leftarrow T_0, x \leftarrow x_0$
while $T > T_f$ **do**
  for $i = 1, \ldots, N_t$ *transitions* **do**
    $x' \leftarrow$ *getNeighbor*(x)
    if $f(x') \leq f(x)$ **then** $x \leftarrow x'$;
    else $x \leftarrow x'$ with probability $\exp \frac{f(x') - f(x)}{T_i}$;
  end
  $T \leftarrow \alpha T$;
end
return X;

The neighborhood operator *getNeighbor*(x) we are proposing for our problem returns a neighbor state $x' = (\rho', \theta', \phi', z', t')$ of the current state $x = (\rho, \theta, \phi, z, t)$ through the following five functions, chosen randomly with, in our tests, an identical probability, 1/5:

- *Translate*(x): return x', where ρ' is chosen randomly in $[0, \rho_{max}]$, and $\theta' := \theta$, $\phi' := \phi$, $z' := z$, $t' := t$.
- *Rot*$_\theta(x)$: Rotate the MA around the military base, return x', where θ' is chosen randomly in $[0, 2\pi]$, and $\rho' := \rho$, $\phi' =: \phi$, $z' := z$, $t' := t$.
- *Rot*$_\phi(x)$: Rotate the MA around its center, return x', where ϕ' is chosen randomly in $[0, 2\pi]$, and $\rho' := \rho$, $\theta' := \theta$, $z' := z$, $t' := t$.
- *ShiftAltitude*(x): return x', where z' is chosen randomly in Z, and $\rho' := \rho$, $\theta' := \theta$, $\phi' := \phi$, $t' := t$.
- *ShiftActivationTime*(x): return x', where t' is chosen randomly in T, and $\rho' := \rho$, $\theta' := \theta$, $\phi' := \phi$, $z' := z$.

All random choices above are assuming a uniform distribution.

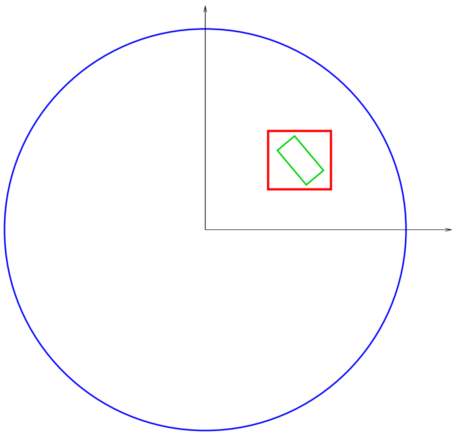

Figure 6: Extract potential affected area (red) in the neighborhood of the current MA (green)

To evaluate the new state, the MA is first located to its new geographical position, and then located in time. Once we have the current 4D position, x, of the MA, one can count the number of trajectory sample points lying inside MA(x). Again, in order to avoid checking each point which would involve excessive amount of calculation, we propose proceeding as follows. First, discretize the 3D airspace and time to construct a 4D grid. Then, extract the neighbor grid elements around the MA as illustrated in red on Fig. 6; these grid elements envelop the MA (green) completely. Trajectory sample points lying outside these grid elements are not inside the MA. It remains to consider the trajectory sample points lying within the (red) envelop. We apply the winding number algorithm to check if the points are inside the MA shape. The winding number algorithm is a computationally-efficient procedure used to determine whether a given 2D point lies inside a given 2D polygon [6], based on the sum of the angles subtended by each side of the polygon.

4 Simulation results in the French airspace

In this section, we apply the simulated annealing algorithm on one day of French traffic involving 8,836 civil flight trajectories, discretized into $M = 1,851,029$ trajectory sample points at intervals of 15 seconds. We first present the input MA parameters used in the numerical tests and the values we choose for the user-defined parameters involved in the SA algorithm. Then, numerical results obtained are presented and analyzed.

The chosen input MA parameter values are listed in Table 1:

Table 1: Input MA parameter values

parameter	value	unit
length (L)	40	NM
width (W)	20	NM
height (H)	5,000	feet
z_{min}	10,000	feet
z_{max}	30,000	feet
δz	1,000	feet
J	20	-
D	120	minutes
t_{min}	10 AM	-
t_{max}	12 AM	-
δt	5	minutes
K	24	-

The SA algorithm parameters are set as follows:

- Cooling parameter: $\alpha = 0.95$
- Number of transitions for each temperature: $N_t = 200$
- Initial temperature: T_0 calculated empirically to yield an 80% acceptance rate
- Final temperature: $T_f = 0.0001T_0$

Table 2: Simulation results with length $L = 40$NM for different values of ρ_{max}

	Case 1	Case 2	Case 3
ρ_{max} (NM)	100	200	400
$nbTrajs$	1,339	1,778	2,113
$nbPts$	100,741	171,331	226,316
$\overline{f(x^*)}$	53.2	51.1	50.8
$\overline{nbPts^*}$	0	0	0
$\Delta nbPt^*$	0	0	0
$\overline{\rho^*}$ (NM)	53.2	51.1	50.8
$\Delta\rho^*$ (NM)	2.3	3.8	1.4
CPU (s)	114	299	285

We first apply the algorithm for three different values of the range ρ_{max}: 100 NM, 200 NM, and 400 NM. Table 2 indicates the number of trajectories ($nbTrajs$) and sample points ($nbPts$) that can potentially be impacted for each case. The SA algorithm was programmed in standard Java language. The results presented in this section are obtained on a computer with an Intel i7-4700MQ 2.40 GHz processor with 4 cores. We run the SA algorithm 10 times with the same parameter setting; average results will be computed over these 10 runs.

In each case, the algorithm finds a location of the MA with the minimal possible impact on the civil flights: zero impact on the civil flights. The maximum solving time of ≈ 2 minutes is viable in a real-time operational context. The results obtained are presented in Table 2, where $\overline{f(x^*)}$ is the average minimal objective function (cf. equation (1)), which is the sum of $\overline{nbPts^*}$, the average minimal number of impacted trajectory sample points (equal to $1_{P_i \in \mathrm{MA}(x^*)}$) and of $\overline{\rho^*}$, the average minimal distance of the MA found by the SA algorithm. The average deviation of $nbPts^*$ for 10 runs is denoted $\Delta nbPt^*$, $\Delta\rho^*$ is the average deviation of ρ^* and CPU represents the average SA algorithm run time in seconds.

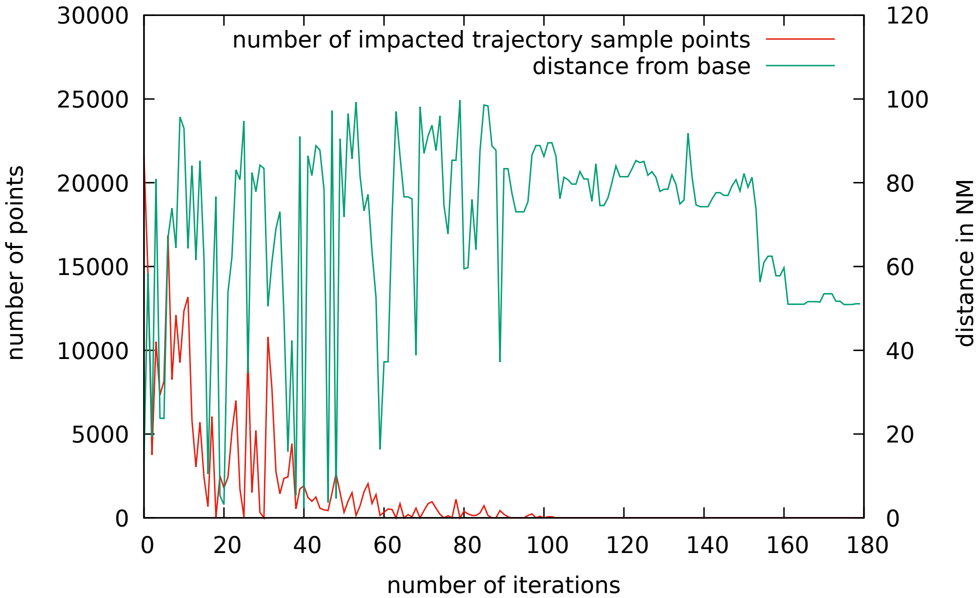

Figure 7: Evolution of the two criteria for $\rho_{max} = 100$ NM

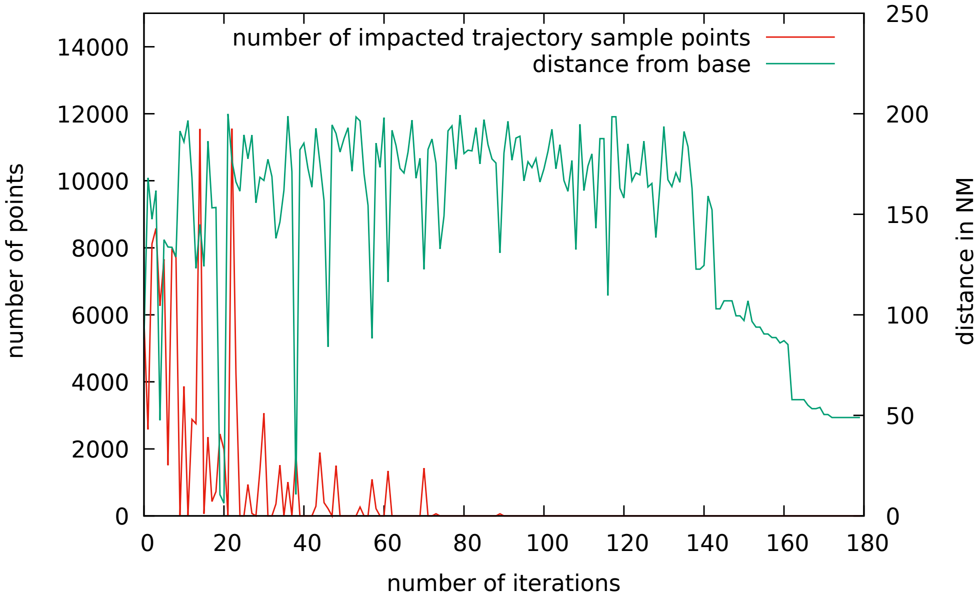

Figure 8: Evolution of the two criteria for $\rho_{max} = 200$ NM

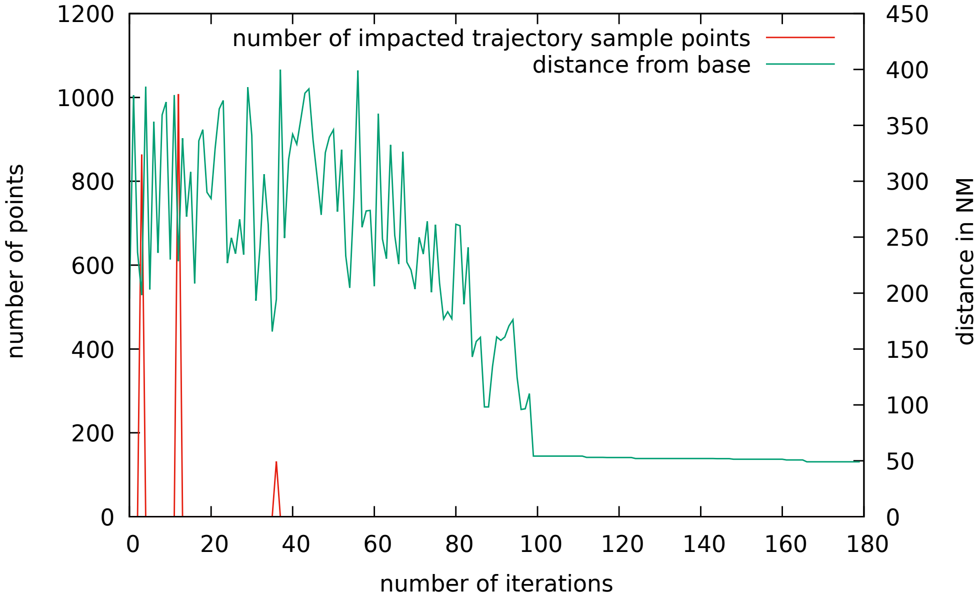

Figure 9: Evolution of the two criteria for $\rho_{max} = 400$ NM

Fig. 7, 8 and 9 display the number of impacted trajectory sample points (first term of objective function (1) in red) and the distance between the MA (second term of objective function (1) in green) and the base at each iteration of the SA algorithm for three different ranges ρ_{max}. In the objective function, the second criterion (distance from the military base, expressed in NM) is a small value in comparison with the first criterion (number of impacted points), since we chose to set the weighting parameter to $\mu = 1$. As a consequence, we observe that the SA algorithm concentrates first on minimizing the first criterion (impact), and then pursues with minimizing the distance from the base while keeping the first criterion at its minimal value. The nearest MA found is 51 NM away from the base for the range 100 NM. The objective-function value converges faster to zero-impact solutions with the increasing range limits, and the nearest MA found is situated at 49 NM from the base for ranges 200 NM and 400 NM. Indeed, the SA algorithm often finds better locations because a larger range allows SA to converge rapidly to zero-impact solution, concentrating thereafter more time on minimizing the distance.

To analyze whether the algorithm is sensitive to the shape and size of the MA, we test different shapes and sizes of the MA by modifying the input parameters: length, width, height and activation duration. The results obtained by SA for shape change and size reduction of the MA are similar to those obtained above. However, if we increase the size of the MA, SA may fail to find an MA with zero impact on civil trajectories under small range limitations. Table 3 displays the results obtained for example when doubling the length of the MA; one observes that $\Delta nbPt^*$ and $\Delta\rho^*$ are relatively large for the 100 NM range since for 3 runs out of 10, the algorithm finds an MA location near the base but impacting hundreds of trajectory sample points.

Table 3: Simulation results with length $L = 80$NM for different values of ρ_{max}

	Case 1	Case 2	Case 3
ρ_{max} (NM)	100	200	400
$nbTraj$	1,561	1,932	2,117
$nbPt$	131,272	188,288	229,178
$\overline{f(x^*)}$	146.2	104	93.1
$\overline{nbPt^*}$	62.4	0	0
$\Delta nbPt^*$	88	0	0
$\overline{\rho^*}$ (NM)	83.8	104	93.1
$\Delta\rho^*$ (NM)	18.8	2.4	0.18
CPU (s)	281	843	596

5 Conclusion and perspectives

As air traffic increases, military areas become a critical constraint for civilian traffic. More flexibility is requested for airspace management. In this article, we proposed a Dynamic Military Area (DMA) approach, concentrating on the optimal location problem of one DMA within civilian traffic dedicated for short-time military activities. The military area is modeled by a 2D geometry shape, with a cylindrical extension associated to given specified flight levels, that has the flexibility to move both in space and in time. We proposed a constrained-optimization formulation that aims at minimizing both the impact of military activities on civil flights, and the distance between the military area and a given military base. A simulated annealing algorithm was designed to address the obtained black-box optimization problem. Numerical experiments were conducted on instances involving one day of French traffic. The results indicate that the proposed methodology is a viable decision-aid tool to search for the nearest military area having no impact on the traffic, as in our tests none of the existing 8,836 civil flights were impacted under different range limitations. Various tests on different input parameters show that the SA algorithm is robust and not significantly sensitive to the shape and size changes of the military area.

In future work, one may address another *dynamic* versions of this location problem, in which the position of the center of the military area follows a given mission trajectory. Another promising track of research could envisage using a deterministic black-box optimization methods such as those proposed in [7, 8]. The military location problem may be combined with a real-time flight-trajectory conflict prediction and resolution problem. In this case, conflict resolution approaches will be applied on the trajectories impacted by the chosen military area. A further extension of this work would involve several types of military aircraft operating from different bases. Finally, the concepts introduced in this study can be applied for UAV (Unmanned Aerial Vehicle) planning in the civilian traffic, or to any other dynamic restricted airspace planning.

6 References

[1] "Airbus global market forecast for 2017-2035", Airbus Group, 2017. URL: http://www.airbus.com /aircraft/market/global-market-forecast.html.

[2] Moradi, E. and Bidkhori, M.,"Single facility location problem", in Facility Location: Concepts, Models, Algorithms and Case Studies, Zanjirani, F. R., and Hekmatfar, M. (editors), Physica-Verlag HD, Heidelberg, 2009. p.37-68.

[3] Plastria, F., "Continuous location problems: Research, results and questions." In Facility location: A survey of applications and methods, Z. Drezner (editor), Springer, 1995. p.85-127.

[4] Brimberg, J., Hansen, P., Mladenovic, N., and Salhi, S., "A survey of solution methods for the continuous location allocation problem" International Journal of Operations Research (2008): 1-12.

[5] "Simulated Annealing: Theory and Applications", Laarhoven, P. J. M. and Aarts, E. H. L. (editors), Kluwer Academic Publishers, Norwell, MA, USA, 1987.

[6] Hormann, K. and Agathos, A., "The point in polygon problem for arbitrary polygons", Computational Geometry, Volume 20, Issue 3, 2001. p. 131-144.

[7] Conn, Andrew R., Scheinberg, Katya, and Vicente, Luis N., "Introduction to Derivative-Free Optimization", Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2009.

[8] Audet, C., and Hare, W., "Derivative-Free and Blackbox Optimization", Springer Series in Operations Research and Financial Engineering, Springer International Publishing, 2018.