Air Traffic Control Approach using Distributed CSPs

Abdellah IDRISSI   El Houssine BOUYAKHF

Research Group in Artificial Intelligence and Pattern Recognition
(GRIARF), Fac. Sciences, Phy. Dep., BP 1014, Av. Ibn Battouta,
University Mohammed V, Agdal, 10000 Rabat, Morocco

idrissi_abd@yahoo.fr   bouyakhf@fsr.ac.ma

Abstract

The Constraints Satisfaction Problems (CSPs) have a great role in the Artificial Intelligence domain. They allow to represent in a simple form a great number of real problems. This is the framework of our contribution. It consists to propose a distributed CSP representation of air traffic conflicts and to propose a methodology of resolution to this problem. In this perspective, we first propose an amelioration of a description-resolution expertise within the framework of formalizing the air traffic controllers way of thinking by a system of production rules (SPR). Then, we deduce distributed constraints from the inference engine results that we have adapted to the SPR. This leads us to the definition of a distributed CSP related to air traffic conflicts problem, to which we propose a resolution method.

key words

Artificial Intelligence, CSP, Inference engine, Air traffic control.

1- Introduction

The saturation of the aerial space is the major problem that the air traffic controllers must face nowadays. The aerial traffic will greatly increase in the next years and regulation problem will stay crucial. At first, we'll briefly describe the way aerial circulation operates, then we'll present aerial conflicts definition and how we'll proceed for the resolution.

Airplanes take off from the departure aerodrome and get to their destination aerodrome by taking a controlled aerial space [15]. Inside this space, security i.e. collisions prevention is not assured by pilots who are not frequently informed about the nearby traffic, but by air traffic controllers. They have detection means and have under their eyes the totality of the traffic. The controlled aerial space is divided into three dimensional defined control sectors. Each control sector is under responsibility of a team of air traffic controllers (two in general). An air traffic controllers team cannot effectively manage the traffic unless its work charge remains under certain limits [14]. Nevertheless, the work charge increases with the number of aircraft inside the sector. The main purpose of air traffic control is security. Thus, an air traffic controllers team won't accept in its sector more than a certain number of airplanes per hour: this number is called sector capacity [7] it's defined for each sector during negotiations between syndicates and direction [2].

Definitions

Separation: We define a horizontal distance expressed in nautical miles (nm): the horizontal separation and the vertical distance expressed in feet (ft): the vertical separation. We say that two airplanes are separated when the distance that separates their projections on a horizontal plan is greater than horizontal separation or when the difference of their altitudes is greater than vertical distance.
Conflict: Two airplanes are said in conflicts when they aren't separated.

We present this work in the framework of resolving the aerial conflicts problem. In this purpose, we propose a resolution approach for aerial conflicts based on artificial intelligence concepts and especially constraints satisfaction problems (CSPs). This approach is divided into two steps : the first concerns the formulation of aerial conflicts problem as a distributed CSP and the second concerns the adaptation of a global resolution method to its resolution.

This paper contains the following : section 4 is reserved to the formulation of the air traffic control problem as a CSP. Before that, we'll describe in section 2 a conflicts and detection-resolution expertise and we'll present in section 3 some notions on CSPs. Then, we'll present in section 5 the resolution algorithm: its technique (section 6) and the obtained results (section 7). Finally, we'll conclude in section 8 including some perspectives for this work.

2- Adaptation of Constraints Satisfaction Problems to Aerial Conflict Problems

In order to represent aerial traffic problem as a CSP, we have chosen first to identify variables and their domains of values, then define problem constraints. Hence we used a description-resolution expertise based on first order predicates logic formalism in order to model conflicts situations and air traffic controllers [12]. In the framework of studying resolution rules used by air traffic controllers, researchers are interested in the construction of geometrical figures composed of two horizontal trajectories in conflicts [12]. These two trajectories are constituted either by two intersecting segments in a common point called cross or by three segments intersecting in two points hence forming central segment called cross segment (figure 1). A detailed study of these trajectories is found in [8].

Production rules have the following form :

Spacial Temporal Description of a Conflict → Action(s).

Controllers way of thinking in order to solve aerial conflicts concerns conflict identification which consists of knowing its geometrical and temporal nature.

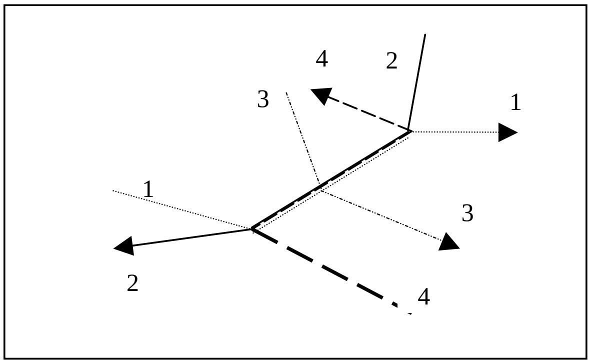

Figure 1: conflicts example to solve

Then experts (controllers) intervene with respect to their vision, by pertinent problem resolution actions. Once conflict nature detected, controllers decide and make some adequate actions. These identification and actions rules are just used for given conflicts, consequently this expertise is deterministic. Actions described in this expertise are in the form of conflict resolution maneuvers between airplanes two by two. These maneuvers that form conclusion part of the rules will be considered as distributed constraints of the aerial traffic CSP. Remember that CSPs make it possible to represent simply a great number of real problems. A CSP is defined by a set of variables associated to a finite domain of discrete values and a set of constraints linking these variables and defining the set of values combinations satisfying the constraint. A solution is an instantiation of variables satisfying all constraints.

3- Solutions search

In order to solve a CSP, it exists in the literature different approaches and methods (Backtrack, Repair, Local changes, Conflict-Directed-back-jumping, Nogood-Recording, Dynamic-backtracking…) [16]. During the time, one or several constraints can be added or removed according to the problem's nature (dynamic nature). This change can come from user's environment or more generally from other agents in a distributed system. Solving a dynamic CSP (DCSP) consists of finding and solving a CSP sequence where each sequence element differs from the precedent by the addition and/or remove of constraints. The problem is to develop methods allowing to maintain solutions and/or deductions whatever is the nature of change. Remember that the six methods mentioned above are all adapted to the dynamic CSP case [16].

4- Representation of Aerial Conflicts Problem (ACP) as a CSP

In the preceding section, we mentioned some CSP methods developed by researchers in artificial intelligence within divers. In this paragraph, we'll propose a link between constraints satisfaction problems (CSPs) and aerial conflicts problems (ACPs).
Before starting solutions search of an ACP like problem by CSP methods, one must identify its variables, domains and constraints linking its variables. These constraints must be binary and distributed between agents (airplanes). Remember that a binary constraint is a subset of the Cartesian product dom[i] × dom[j] that gives couples of values that can be taken by variables i and j; and a solution is an instantiation of variables that violates no constraint. Note that a constraint linking two variables i and j can be written $((i, a), (j, b)) \in Cij$ which specifies that the constraint allows having value "a" for variable "i" and "b" for "j".

In this context, we found useful to exploit production rules mentioned in the preceding paragraph. the premise part will allow the identification of relations between variables (airplanes) and conclusion part will be considered as a set of allowed values for variables. In a nutshell, we'll consider a set of variables $A = \{1,2,\ldots,n\}$ representing the set of airplanes, where 1,2,…,n are identifiers of airplanes; C^2_n represents the number of airplanes couples in conflicts ; $Di = \{d1,\ldots, dn, g1\ldots, gn, p, h1,\ldots, hn, b1,\ldots, bn\}$, domain of variable airplane i .

We must note that we enriched the description presented in [12] by :
- Extension of the set of possible positions that can an airplane "i" take with respect to another "j". Indeed, predicates POSITION(i, m, j) with $m \in M' = \{G, D\}$ where G, D mean respectively that airplane i is in left position (resp right) with respect to airplane j, will be extended by considering the new $M' = \{G, D, H, B\}$, where H (resp B) mean that airplane i is in top (resp in bottom) with respect to airplane j [9]. Thus, we'll have (4p-2) predicates to add to the control system knowledge base: POSITION(i, D_1, j),…, POSITION(i, D_p, j) ; POSITION(i, G_1, j),…, POSITION(i, G_p, j) ; POSITION(i, H_1, j),…, POSITION(i, H_p, j) ; POSITION(i, B_1, j),…, POSITION(i, B_p, j) showing respectively that airplane i is in the right, in the left, above, under one or p sector(s) of the airplane j;

- Maneuvers domain extension which operates just on the horizontal plan; instead of MANEUVER(i, m) which indicates that airplane i can be maneuvered to m, with $m \in M = \{G, D, P\}$ where G is airplane i maneuver to the left, D to the right and P indicates that the airplane persists on its rectilinear trajectory; we considered that m can take all values of Di. In other terms, we also took into consideration vertical plan;
- Enlargement of the conclusion part of resolution actions by considering that predicate MANEUVER(i, D) can be replaced by a clause of predicates MAN(i, m) such that $m \in D \cup H \cup B$. In the same way, predicate MANEUVER(i, G) will be replaced by a clause of predicates MAN(i, m') such that $m' \in G \cup H \cup B$, with D = {d1,…, dn}, G = {g1,…, gn}, P = {p}, H = {h1,…, hn} and B = {b1,…, bn}; di, gi, hi and bi represent respectively a deviation of the airplane in order to locate on the i^{th} sector which is just in the right, in the left, on top or in bottom. For a more detailed description of these changes, see [9].

Now we identify these resultant maneuvers of resolution actions of distributed constraints airplanes couple in question ;

- Transformation by a rotation of $\pi/2$ of all conflicts situations and resolution maneuvers situated in the horizontal plan in order to relocate in the vertical plan. This permitted us to add 21 situations and 12 rules or axioms to the knowledge base of the quoted expertise. In total, we have 42 situations and 24 axioms. These extensions was obtained by substituting "h" by "d" and "b" by "g" in the predefined situations. For example MAN(i, d_n) will be replaced by MAN(i, h_n) ;
- Augmentation of constraints number between a couple of airplanes in conflict. This number is obtained with an increase by one of the number of disjunction which is found in the resolution, the latter must be besides presented in a normal disjunctive form. A conjunction, between two predicates MAN(i, x) and MAN(j, y) with $x, y \in Di$, corresponds to a lonely constraint.

Until now, we highlighted conflicts resolution between airplanes two by two, but compatibility is not guaranteed between different resolutions. In other words, it can happen that resolution Rjk of the conflict between two airplanes j and k, disrupts the resolution Rij already existing between the two airplanes i and j. In order to offset this problem, we can call at this level one of the algorithms DnAC6, branch and bound or forward-checking, so as to insure coherence or arc consistency adapted to the case of dynamic CSP. One can refer to [3] for a precise study of the algorithm DnAC6, to do likewise refer to [10] for the two others.

Furthermore, given that variables (airplanes), domain of possible values for these variables (Di = {d1,…, dn, g1,…, gn, p, h1,…, hn, b1,…, bn}) and the constraints Cij distributed between agents (airplanes i and j) [Cij = ((i, m), (j, m')) note Cij = (m, m') with $m, m' \in Di$] are well identified; it remains to apply one of solutions search. We opted for the adaptation of Backtrack approach presented in [10, 16].

5- Algorithm of aerial conflicts problems

This algorithm is decomposed into 5 pseudo-codes Pi :

```
P1 ♦ A = {1,2,…,n}, set of  variables (airplanes) ;
     ;;  n = number of airplanes
   ♦ Da = {d1,…, dn, g1,…, gn, p, h1,…, hn, b1,…, bn},
       domain of airplane a
     ;; in order to simplify, we confound all other
     ;; domains with Da
   ♦ $C^2_n$ = number of airplanes couples in conflicts
```

```
P2 ;; Decomposition of conflicts between n airplanes to
   ;; couples of conflict ($a_i$, $a_{i+k}$)
 ♦ for i = 1  to  n – 1
       for k = 1  to  n – i
          consider only couples of airplanes having the
          form ($a_i$, $a_{i+k}$)
       end for
   end for
```

```
P3 ;; Resolution of conflict between each couple
   ;; separately by application of resolution axioms
   ;; (enhanced expertise : inference engine [9])
♦ for $a_i, a_j \in A$, with $i \neq j$, only $a_i$ or $a_j$ is maneuvered and
      the other persists on its initial trajectory
          if no indication of maneuver on $a_i$
               then  maneuver ($a_i$, p)  is taken by default
          end if
   end for
```

```
P4 ;; each resolution Rij of conflict CF between a couple
   ;; of airplanes ($a_i$, $a_j$)   represents a set of
   ;; distributed constraints  between airplanes
♦ number of constraints = number of disjunction + 1
```

```
P5 ;; Appeal to procedure dynamic Backtrack in order to
   ;; end up at a global and coherent resolution of
   ;; conflicts between  n  airplanes.
♦   BT(CF)
        return BT-AIRPLANES (∅, AIRPLANES (CF))
    •  BT-AIRPLANES (A1, A2)
       ;; A1 is the set of  affected airplanes.
       ;; A2 is the set of non affected airplanes
        if  A2 = ∅
           then return success
        else $a_i$ = AIRPLANE-CHOICE (A2)
           return BT-AIRPLANE (A1, A2, $a_i$, DOMAIN
           ($a_i$))
    •  BT-AIRPLANE (A1, A2, $a_i$, d)
        if d = ∅
           then return failure
        else  man = MAN-CHOICE (d)
              if BT-MAN (A1, A2, $a_i$, man) = success
                 then  return success
              else  return BT-AIRPLANE (A1, A2, $a_i$,
                d - {man})
```

```
• BT-MAN (A1, A2, $a_i$, man)
  ASSIGN ($a_i$, man)
      if UNSATISFIED–CONSTRAINT
          (CONSTRAINTS ($a_i$) , $A1 \cup \{man\}$)
              then UNASSIGN ($a_i$)
              return failure
      else if BT-AIRPLANES($A1 \cup \{a_i\}$, $A2-\{a_i\}$)
                      = failure
                      then UNASSIGN ($a_i$)
                      return failure
             else return success
```

Figure 2 : Algorithm of aerial conflicts resolution

Non explicit functions in the algorithm are :

- AIRPLANES (CF) returns the set of airplanes of conflict CF ;
- AIRPLANE-CHOICE (A) returns an airplane from the set of airplanes A ;
- DOMAIN (a_i) returns the set of possible maneuvers for airplane a_i ;
- MAN-CHOICE (d) returns a chosen maneuver in the set of maneuvers d ;
- CONSTRAINTS (a_i) returns the set of constraints that influence the variable airplane a_i ;
- UNSATISFIED-CONSTRAINT (C, A) returns true if a constraint of the set of constraints C is not satisfied because of the affectation of the set of airplanes A and returns false otherwise ;
- ASSIGN (a_i, man) affects maneuver man to airplane a_i ;
- UNASSIGN (a_i) disaffects airplane a_i .

6- Technique of resolution

The execution of this algorithm of the air traffic control problem resolution is done in two steps :

step 1 :

It consists of identification of variables and their domain (P1) and of realization of an inference engine (P2 and P3) that allows to solve conflicts between airplanes two by two. These resolutions will be taken in the form of distributed constraints (P4) so as to define a distributed CSP dealing with air traffic problem. We proposed an implementation in Le-Lisp for the inference engine. The latter allows to draw a rule from the 24 proposed [9] in the control system knowledge base.

step 2 :

It consists of searching for one (or several) solution(s) to insure global resolution of conflicts between n airplanes in question. This task is accomplished by Backtrack algorithm (P5). The latter considers produced results by the inference engine as distributed constraints between airplanes of its predefined CSP. An implementation, in Le-Lisp of this algorithm, is mentioned in the first part of annex 2 [9]. Variables are airplanes and domains of values are different possible maneuvers of which dispose aerial company operating with this method of resolution. We mentioned hereafter in figure 3 an explicative schema of the principle of proposed algorithm.

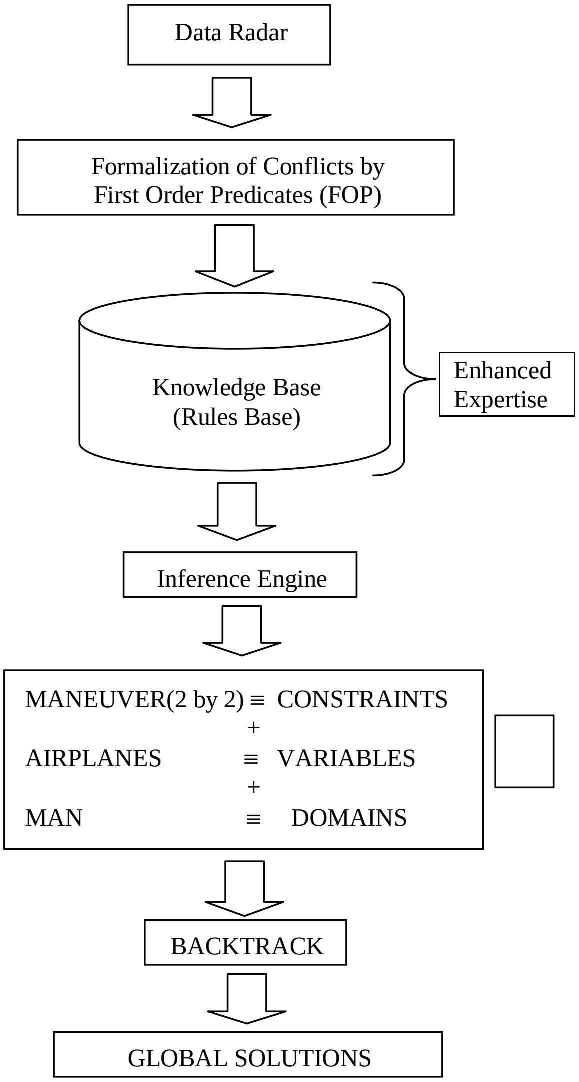

Figure 3 : Synoptic schema of the algorithm

7- Results

The prototype is coded in Le-Lisp on a Personal Computer Pentium II of 333 MHz and 64 MB of RAM. The number of known situations, of conflicts between airplanes two by two, is equal to 42. 21 in horizontal plan and 21 in vertical plan. The number of axioms or resolution rules is 24. The number of constraints between two airplanes in conflict is equal to the number of disjunctions, of the conclusion part of the rule, increased by one. The choice of a constraint taking into consideration an orientation in the vertical plan can only be done in the case of lack of possibilities of an orientation in the horizontal plan.

In order to simplify, we supposed that aerial company dispose of only one sector in the right "d", one in the left "g", one at the top "h" and another in the bottom "b", during conflict detection, on which the airplane can be turned in relation to its initial position "p". In other words, $Da_i = \{d, g, p, h, b\}$. We applied this resolution method to the example of the four airplanes in conflicts ($A = \{1, 2, 3, 4\}$) presented in section 1 (figure 1). The number of constraints obtained by inference engine for each of the 6 couples of airplanes in conflicts (1, 2), (1, 3), (1, 4), (2, 3), (2, 4) and (3, 4) is given by the table of figure 4.

Couple	Number of Constraints
(1, 2)	19
(1, 3)	10
(1, 4)	10
(2, 3)	18
(2, 4)	20
(3, 4)	13

Figure 4 : Number of constraints by couple of conflict

Example of formalization and resolution

Let's take $A = \{1, 2, 3, 4\}$, $n = 4$ airplanes.

Trajectories of airplanes 1, 2 and 4 are each constituted by three broken segments. Trajectory of airplane 3 is composed by two segments : figure 5.

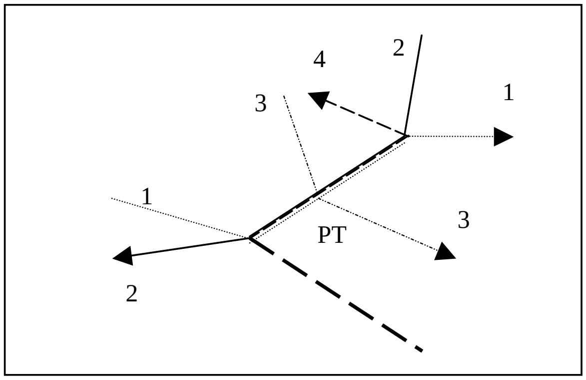

Figure 5 : Example of conflict to be solved

We suppose that airplane 1 arrives, temporarily, to the cross point before airplane 3 and that it arrives in the segment before airplane 2. Airplanes 2, 3 and 4 arrive simultaneously to the cross point.

Logical representation of conflicts

Six couples of airplanes in conflicts ($n = 4 \rightarrow C^2_6 = 6$) : (1, 2), (1, 3), (1, 4), (2, 3), (2, 4) and (3, 4).

Couple (1, 2) is a cross on one segment with overstepping of trajectories and in inverse direction. We have predicates:
CROISEMENT(1, 2) and SENS(1, 2, S, SI).

Couple (1, 3) is a cross in one point with overstepping of trajectories. We have predicates :
CROISEMENT(1, 3) and SENS(1, 3, PT, MS).

Couple (1, 4) is a cross on one segment with overstepping of trajectories and in the same direction. We have predicates :
CROISEMENT(1, 4) and SENS(1, 4, S, MS),

Couple (2, 3) is a cross on one point with overstepping of trajectories. We have predicates :
CROISEMENT(2, 3) and SENS(2, 3, PT, s).

Couple (2, 4) is a cross on one segment with overstepping of trajectories and in inverse direction. We have predicates :
CROISEMENT(2, 4) and SENS(2, 4, S, SI),

Couple (3, 4) is a cross on one point with overstepping of trajectories. We have predicates :
CROISEMENT(3, 4), SENS(3, 4, PT, s) and TOURNE(4, D)

According to figure 5, relative positions of airplanes are :
POSITION(1, D, 2), POSITION(1, D, 3), POSITION(1, G, 4), POSITION(2, D, 3), POSITION(2, D, 4) and POSITION(3, G, 4).

Temporal predicates are :
PREMIER(1, 2), PREMIER(1, 3), PREMIER(1, 4), SIMUL(2, 3), SIMUL(2, 4) and SIMUL(3, 4).

Note

For couple (3, 4) we made a transformation of trajectory of the airplane 4 considering that it can be composed in this case of two broken segments instead of three. This is because trajectory of airplane 3 is composed only of two segments and consequently conflict is localized in the central point PT.

Distributed Constraints : Inference Engine Results

Couple (1, 2)

CONFLICT(1, 2) ∧ POSITION(1, D, 2) ∧ SENS(1, 2, S, SI) ∧ CROISEMENT(1, 2) ∧ PREMIER(1, 2) →
" MAN i d ∧ MAN j p " ∨ (1)
" MAN i d ∧ MAN j h " ∨ (2)
" MAN i d ∧ MAN j b " ∨ (3)
" MAN i h ∧ MAN j p " ∨ (4)
" MAN i h ∧ MAN j h " ∨ (5)
" MAN i h ∧ MAN j b " ∨ (6)
" MAN i b ∧ MAN j p " ∨ (7)
" MAN i b ∧ MAN j h " ∨ (8)
" MAN i b ∧ MAN j b " ∨ (9)
" MAN i g ∧ MAN j p " ∨ (10)
" MAN i g ∧ MAN j h " ∨ (11)
" MAN i g ∧ MAN j b " ∨ (12)
" MAN i p ∧ MAN j d " ∨ (13)
" MAN i h ∧ MAN j d " ∨ (14)
" MAN i b ∧ MAN j d " ∨ (15)
" MAN i p ∧ MAN j h " ∨ (16)
" MAN i b ∧ MAN j h " ∨ (17)
" MAN i p ∧ MAN j b " ∨ (18)
" MAN i h ∧ MAN j b " (19)
(axiom A6 [9])

Couple (1, 3)
CONFLICT(1, 3) ∧ POSITION(1, D, 3) ∧ SENS(1, 3, PT, MS) ∧ CROISEMENT(1, 3) ∧ PREMIER(1, 3) →
" MAN 1 p ∧ MAN 3 d " ∨ (1)
" MAN 1 p ∧ MAN 3 h " ∨ (2)
" MAN 1 p ∧ MAN 3 b " ∨ (3)
" MAN 1 d ∧ MAN 3 d " ∨ (4)
" MAN 1 d ∧ MAN 3 h " ∨ (5)
" MAN 1 d ∧ MAN 3 b " ∨ (6)
" MAN 1 b ∧ MAN 3 d " ∨ (7)
" MAN 1 b ∧ MAN 3 h " ∨ (8)
" MAN 1 h ∧ MAN 3 d " ∨ (9)
" MAN 1 h ∧ MAN 3 h " (10)
(axiom A1 [9])

Couple (1, 4) :
CONFLICT(1, 4) ∧ CROISEMENT(1, 4) ∧ POSITION(1, G, 4) ∧ SENS(1, 4, S, MS) ∧ PREMIER(1, 4) →
" MAN 1 p ∧ MAN 4 g " ∨ (1)
" MAN 1 p ∧ MAN 4 h " ∨ (2)
" MAN 1 p ∧ MAN 4 b " ∨ (3)
" MAN 1 d ∧ MAN 4 g " ∨ (4)
" MAN 1 d ∧ MAN 4 h " ∨ (5)
" MAN 1 d ∧ MAN 4 b " ∨ (6)
" MAN 1 b ∧ MAN 4 g " ∨ (7)
" MAN 1 b ∧ MAN 4 h " ∨ (8)
" MAN 1 h ∧ MAN 4 g " ∨ (9)
" MAN 1 h ∧ MAN 4 h " (10)
(axiom A2 [9])

Couple (2, 3)
CONFLIT(2, 3) ∧ POSITION(2, G, 3) ∧ SIMUL(2, 3) ∧ CROISEMENT(2, 3) ∧ SENS(2, 3, PT, s) ∧ ¬TOURNE(3, G) ∧ TOURNE(2, D) →
" MAN 2 d ∧ MAN 3 p " ∨ (1)
" MAN 2 d ∧ MAN 3 h " ∨ (2)
" MAN 2 d ∧ MAN 3 b " ∨ (3)
" MAN 2 d ∧ MAN 3 g " ∨ (4)
" MAN 2 h ∧ MAN 3 d " ∨ (5)
" MAN 2 h ∧ MAN 3 b " ∨ (6)
" MAN 2 h ∧ MAN 3 p " ∨ (7)
" MAN 2 h ∧ MAN 3 g " ∨ (8)
" MAN 2 b ∧ MAN 3 b " ∨ (9)
" MAN 2 b ∧ MAN 3 p " ∨ (10)
" MAN 2 b ∧ MAN 3 h " ∨ (11)
" MAN 2 b ∧ MAN 3 d " ∨ (12)
" MAN 2 p ∧ MAN 3 d " ∨ (13)
" MAN 2 p ∧ MAN 3 h " ∨ (14)
" MAN 2 p ∧ MAN 3 b " ∨ (15)
" MAN 2 p ∧ MAN 3 g " ∨ (16)
" MAN 2 d ∧ MAN 3 d " ∨ (17)
" MAN 2 h ∧ MAN 3 h " (18)
(axiom A3 [9])

Couple (2, 4) :
CONFLICT(2, 4) ∧ CROISEMENT(2, 4) ∧ POSITION(2, D, 4) ∧ SENS(2,4, S, SI) ∧ SIMUL(2, 4) →
" MAN 2 d ∧ MAN 4 p " ∨ (1)
" MAN 2 d ∧ MAN 4 h " ∨ (2)
" MAN 2 d ∧ MAN 4 b " ∨ (3)
" MAN 2 d ∧ MAN 4 g " ∨ (4)
" MAN 2 h ∧ MAN 4 d " ∨ (5)
" MAN 2 h ∧ MAN 4 b " ∨ (6)
" MAN 2 h ∧ MAN 4 p " ∨ (7)
" MAN 2 h ∧ MAN 4 g " ∨ (8)
" MAN 2 b ∧ MAN 4 g " ∨ (9)
" MAN 2 b ∧ MAN 4 p " ∨ (10)
" MAN 2 b ∧ MAN 4 h " ∨ (11)
" MAN 2 b ∧ MAN 4 d " ∨ (12)
" MAN 2 p ∧ MAN 4 d " ∨ (13)
" MAN 2 p ∧ MAN 4 h " ∨ (14)
" MAN 2 p ∧ MAN 4 b " ∨ (15)
" MAN 2 p ∧ MAN 4 g " ∨ (16)
" MAN 2 g ∧ MAN 4 d " ∨ (17)
" MAN 2 g ∧ MAN 4 h " ∨ (18)
" MAN 2 g ∧ MAN 4 b " ∨ (19)
" MAN 2 g ∧ MAN 4 p " (20)
(axiom A5 [9])

Couple (3, 4) :
CONFLICT(3, 4) ∧ CROISEMENT(3, 4) ∧ POSITION(3, G, 4) ∧ SENS(3, 4, PT, s) ∧ SIMUL(3, 4) ∧ TOURNE(3, G) ∧ TOURNE(4, D) →
" MAN 3 d ∧ MAN 4 g " ∨ (1)
" MAN 3 d ∧ MAN 4 h " ∨ (2)
" MAN 3 d ∧ MAN 4 b " ∨ (3)
" MAN 3 g ∧ MAN 4 g " ∨ (4)
" MAN 3 g ∧ MAN 4 h " ∨ (5)
" MAN 3 g ∧ MAN 4 h " ∨ (6)
" MAN 3 p ∧ MAN 4 g " ∨ (7)
" MAN 3 p ∧ MAN 4 h " ∨ (8)
" MAN 3 p ∧ MAN 4 b " ∨ (9)
" MAN 3 h ∧ MAN 4 g " ∨ (10)
" MAN 3 h ∧ MAN 4 b " ∨ (11)
" MAN 3 b ∧ MAN 4 g " ∨ (12)
" MAN 3 b ∧ MAN 4 h " (13)
(axiom A2 [9])

Thus, we defined a Distributed CSP relative to the case of aerial conflicts, and we adapted to its resolution a global solutions search method "P5". The number of found solutions for the type of conflicts presented in the preceding example after 8059 combinations is 87 which three of them are :
{ MAN(1, d), MAN(2, p), MAN(3, b), MAN(4, h) } ;
{ MAN(1, g), MAN(2, b), MAN(3, h), MAN(4, p) } and
{ MAN(1, h), MAN(2, d), MAN(3, b), MAN(4, p) }.

8- Conclusion

With the intention of helping the aerial controller to take decisions during detection-resolution of conflicts. We studied aerial traffic problems and certain concepts of artificial intelligence, among others constraints satisfaction problems (CSPs). The latter allow to represent in a simple form a great number of real problems. In this scope we proposed a method of resolution of aerial conflicts. But before that, we enriched and enhanced the expertise of description-resolution proposed in [12] by enlarging formalizations of situations and formalizations of actions on the airplanes in the vertical plan. Then we adapted an inference engine that allows to solve conflicts between airplanes two by two. The result of this engine will be considered in the form of a set of distributed constraints that will be used by Backtrack algorithm, so as to solve conflicts problem in its overall. Airplanes are CSP variables adapted to the case of aerial conflicts, maneuvers are values that can be taken by these variables.

Moreover, with the conditions imposed by the model of partial consistency, CSP and DCSP, we can sometimes detect absence of solution even before the exploration of all possibilities. We indicate that the appeal to flexible CSP (FCSP) in order to imply what is called ″preferences or criterions″ instead of ″constraints″ [5] prove to be necessary. In other words, in the absence of an exact solution, a quasi optimal solution have to be searched in order to release the situation using FCSP. It seems to us that FCSP approach looks promising for aerial conflicts problem resolution (ACP). One must also use diverse techniques of representation of time [1], of temporal scheduling [13] and of resources management [11] so as to integrate all of them to the control system knowledge base.

References

[1] J.F. Allen, Towards a general theory of action and time. Artificial Intelligence, vol. 23, pp. 123-154, 1984.

[2] R. Amalberti, Les effets pervers de l’ultra sécurité. Revue la recherche n°319, pp. 66-70. Avril 1999.

[3] R. Debruyne, les algorithmes d’arc-consistance dans les CSP dynamiques. Revue d’Intelligence Artificielle, vol. 9 n°3, pp. 239-267, 1995.

[4] N. Durand et J.M. Alliot, Peut-on supprimer le contrôle au sol ?. Revue la recherche n°319, pp. 57-61. Avril 1999.

[5] H. Fargier, D. Dubois et H. Prade, Problèmes de satisfaction de constraints flexibles : Une approche égalitariste. Revue d’Intelligence Artificielle, vol. 9, n°3, 1995.

[6] J. M. Garot, L’Europe a aussi cédé au mythe du projet Apollo. Revue la recherche n°319, p. 65. Avril 1999.

[7] S. Gruszow, La bataille du carton, du stylo bille et de la souris. Revue la recherche n°319, pp. 62-64. Avril 1999.

[8] A. Idrissi, Formalisation des raisonnements des contrôleurs aériens dans la resolution de conflicts. Rapport interne GRIARF, Intelligence Artificielle. Faculté des Sciences de Rabat. Avril 1999.

[9] A. Idrissi, Approche des Problèmes du trafic aérien par les concepts de l'Intelligence Artificielle: Contribution au développement de méthodes de resolution de conflicts par les CSP. Mémoire du DESA, Sciences de l'Ingénieur, GRIARF, Faculté des Sciences de Rabat. Novembre 1999.

[10] P. Jégou, Contribution à l’étude des problèmes de satisfaction de constraints : Algorithmes de propagation et de resolution, Propagation de constraints dans les réseaux dynamiques. Thèse d’université Montpellier II, sciences et techniques du Languedoc. Janvier 1991.

[11] P. Laborie, Planification et gestion de ressources. Rapport LAAS n° 94077, CNRS. Mars 1994.

[12] V. Dugat et L. Lapasset, Formalisation des raisonnements des contrôleurs aériens dans la resolution de conflicts. Clermont-Ferrand, RFIA 98, pp. 367-376. Janvier 1998.

[13] I. Meiri, Combining qualitative and quantitative constraints in temporal reasoning. In proc. 9th AAAI, An aheim, Ca, pp. 260-267, 1991.

[14] L. Penhouet, L. Zerrouki et R. Fondacci, Modèle de régulation à court terme de la charge des secteurs de l’espace aérien. Modélisation du trafic. Actes du groupe de travail 1996, Arcueil, 1996.

[15] J. B. Stuchlik, Le chaos programmé du ciel européen. Revue la recherche n°319, pp. 52-56. Avril 1999.

[16] G. Verfaillie et T. Schiex, Maintien de solution dans les problèmes dynamiques de satisfaction de constraints : Bilan de quelques approches. Revue d’Intelligence Artificielle, vol. 9, n° 3, pp. 269-309, 1995.