Ant Colony Optimization for Air Traffic Conflict Resolution

Nicolas Durand, Jean-Marc Alliot

DSNA/R&D/POM[1]
http://pom.tls.cena.fr/pom

July 1, 2009

[1] DSNA/DTI R&D/Planing Optimization Modeling Team

Introduction

Ant Colony Optimization

Application to Conflict Resolution

Conclusion

Conflict Resolution

- Current Situation : no effective tool for separating aircraft

Conflict Resolution

- Current Situation : no effective tool for separating aircraft
- New means : GPS capabilities (FMS enhancement), Data-Link communications ⇒ Enhance Trajectory Prediction

Conflict Resolution

- Current Situation : no effective tool for separating aircraft
- New means : GPS capabilities (FMS enhancement), Data-Link communications ⇒ Enhance Trajectory Prediction
- Pairwise conflicts ⇒ Clusters

Conflict Resolution

- Current Situation : no effective tool for separating aircraft
- New means : GPS capabilities (FMS enhancement), Data-Link communications $\Rightarrow$ Enhance Trajectory Prediction
- Pairwise conflicts $\Rightarrow$ Clusters
- $\Rightarrow$ High complexity of the underlying problem

Conflict Resolution

- Current Situation : no effective tool for separating aircraft
- New means : GPS capabilities (FMS enhancement), Data-Link communications $\Rightarrow$ Enhance Trajectory Prediction
- Pairwise conflicts $\Rightarrow$ Clusters
- $\Rightarrow$ High complexity of the underlying problem
- Example : solving a n aircraft conflict in the horizontal plane $\Rightarrow \frac{n(n-1)}{2}$ aircraft pairs $\Rightarrow 2^{\frac{n(n-1)}{2}}$ connected components to explore

Conflict Resolution

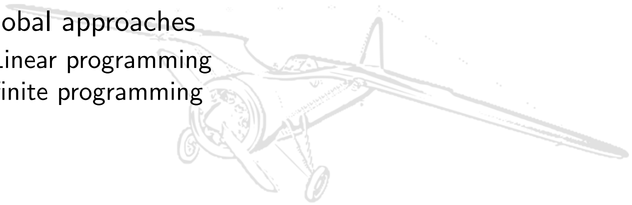
- Current Situation : no effective tool for separating aircraft
- New means : GPS capabilities (FMS enhancement), Data-Link communications $\Rightarrow$ Enhance Trajectory Prediction
- Pairwise conflicts $\Rightarrow$ Clusters
- $\Rightarrow$ High complexity of the underlying problem
- Example : solving a n aircraft conflict in the horizontal plane $\Rightarrow \frac{n(n-1)}{2}$ aircraft pairs $\Rightarrow 2^{\frac{n(n-1)}{2}}$ connected components to explore
- $\Rightarrow$ Local optimization uneffective

Examples of existing algorithms

- Central & Global approaches
  - Integer Linear programming

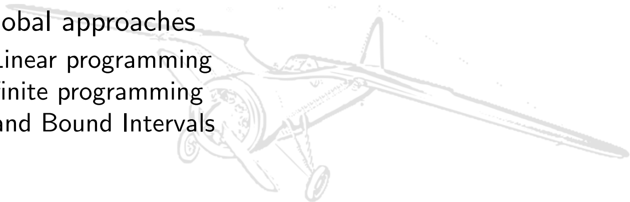
Examples of existing algorithms

- Central & Global approaches
  - Integer Linear programming
  - Semi-definite programming



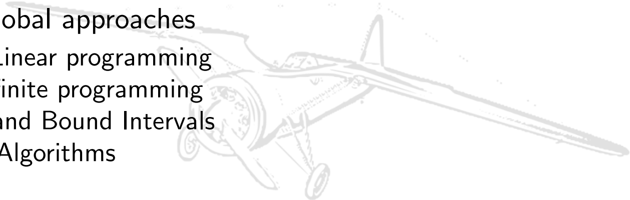
Examples of existing algorithms

- Central & Global approaches
  - Integer Linear programming
  - Semi-definite programming
  - Branch and Bound Intervals



Examples of existing algorithms

- Central & Global approaches
  - Integer Linear programming
  - Semi-definite programming
  - Branch and Bound Intervals
  - Genetic Algorithms



Examples of existing algorithms

- Central & Global approaches
  - Integer Linear programming
  - Semi-definite programming
  - Branch and Bound Intervals
  - Genetic Algorithms
- Autonomous approaches

Examples of existing algorithms

- Central & Global approaches
  - Integer Linear programming
  - Semi-definite programming
  - Branch and Bound Intervals
  - Genetic Algorithms
- Autonomous approaches
  - Neural network

Examples of existing algorithms

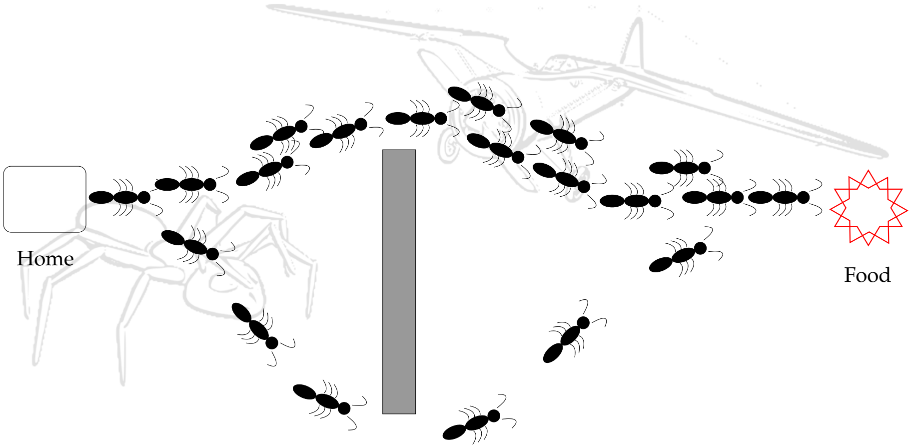
- Central & Global approaches
  - Integer Linear programming
  - Semi-definite programming
  - Branch and Bound Intervals
  - Genetic Algorithms
- Autonomous approaches
  - Neural network
  - Repulsive forces

Examples of existing algorithms

- Central & Global approaches
  - Integer Linear programming
  - Semi-definite programming
  - Branch and Bound Intervals
  - Genetic Algorithms
- Autonomous approaches
  - Neural network
  - Repulsive forces
- Iterative approaches : give priorities to aircraft and use local optimization (for example : A^* algorithm).

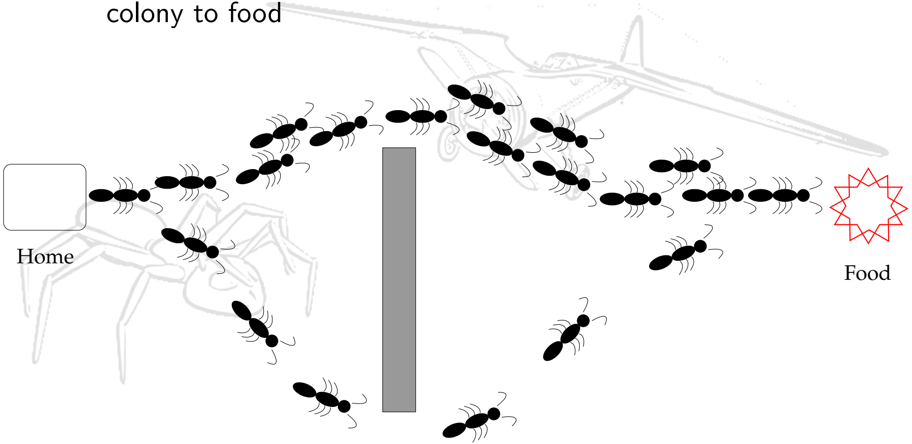
ACO principles

- Use the environment as a medium of communication



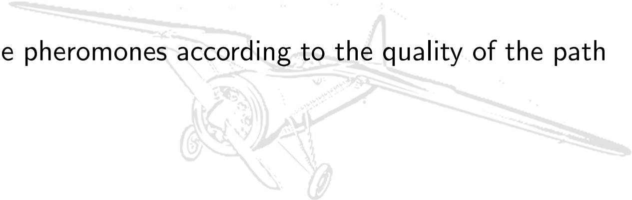
ACO principles

- Use the environment as a medium of communication
- Mimic the ants trying to find the shortest path from their colony to food



ACO algorithm principle

- Ants deposite pheromones according to the quality of the path they find



ACO algorithm principle

- Ants deposite pheromones according to the quality of the path they find
- Ants more likely to follow paths with the most pheromones

ACO algorithm principle

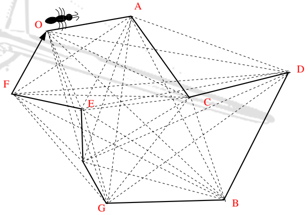
- Ants deposite pheromones according to the quality of the path they find
- Ants more likely to follow paths with the most pheromones
- Add evaporation process to prevent algorithm from local convergence

ACO algorithm principle

- Ants deposite pheromones according to the quality of the path they find
- Ants more likely to follow paths with the most pheromones
- Add evaporation process to prevent algorithm from local convergence
- Stop when no more improvement

ACO for the Traveling Salesman Problem

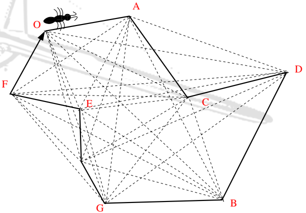
- Ants sent on graph. Each ant builds complete path. Choice of next city influenced by pheromone quantity on paths.



Vidéo

ACO for the Traveling Salesman Problem

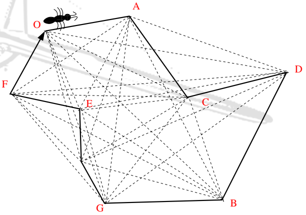
- Ants sent on graph. Each ant builds complete path. Choice of next city influenced by pheromone quantity on paths.
- Ants deposite pheromones on the path chosen : $\Delta\tau_{ij}(t) \propto \dfrac{1}{\sum L_{ij}}$



Vidéo

ACO for the Traveling Salesman Problem

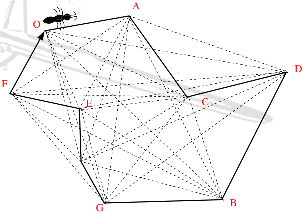
- Ants sent on graph. Each ant builds complete path. Choice of next city influenced by pheromone quantity on paths.
- Ants deposite pheromones on the path chosen : $\Delta\tau_{ij}(t) \propto \dfrac{1}{\sum L_{ij}}$
- At each iteration, evaporate trails : $\tau_{ij} \leftarrow \rho \cdot \tau_{ij}$



Vidéo

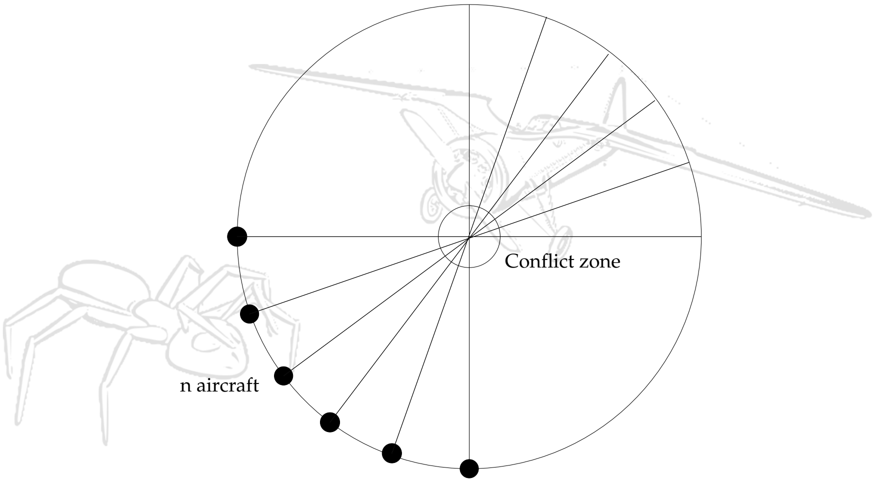
ACO for the Traveling Salesman Problem

- Ants sent on graph. Each ant builds complete path. Choice of next city influenced by pheromone quantity on paths.
- Ants deposite pheromones on the path chosen : $\Delta\tau_{ij}(t) \propto \frac{1}{\sum L_{ij}}$
- At each iteration, evaporate trails : $\tau_{ij} \leftarrow \rho \cdot \tau_{ij}$
- Stop when no more improvement

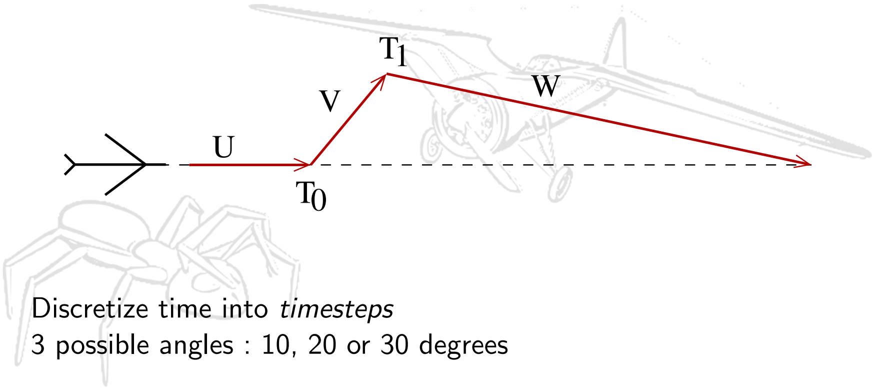


Vidéo

n aircraft conflict example

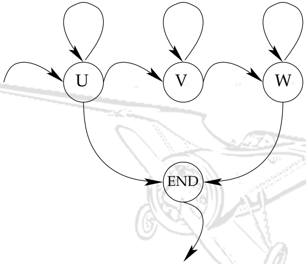


Maneuver modeling



Discretize time into *timesteps*
3 possible angles : 10, 20 or 30 degrees

Possible transitions



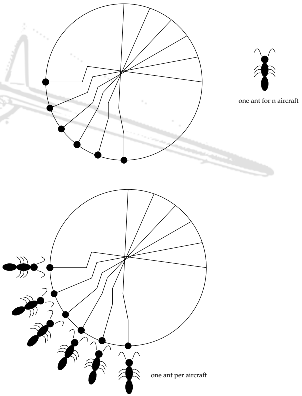
$$U_{i+1} = U_i$$
$$V_{i+1} = V_i + 6$$
$$W_{i+1} = V_i$$

$U_1 = 1,\ V_1 = 6$ and $W_1 = 0$

Number of possible states at timestep $i = U_i + V_i + W_i = 12\,i - 5$

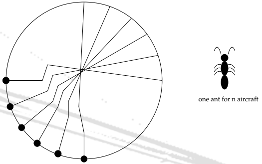
One ant per cluster or one ant per aircraft

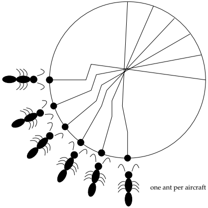
- one ant $\longrightarrow$ one cluster



One ant per cluster or one ant per aircraft

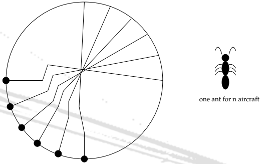
- one ant $\longrightarrow$ one cluster
- for n aircraft and t timesteps : $(12\,t - 5)^n$ trails.

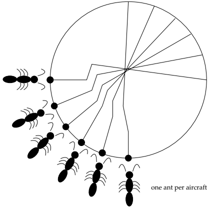




One ant per cluster or one ant per aircraft

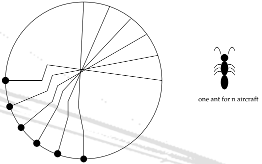
- one ant $\longrightarrow$ one cluster
- for n aircraft and t timesteps : $(12\,t-5)^n$ trails.
- For $n=5$ and $t=10$: more than 10^{10} trails

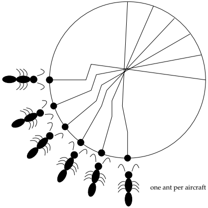




One ant per cluster or one ant per aircraft

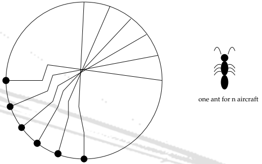
- one ant $\longrightarrow$ one cluster
- for n aircraft and t timesteps : $(12\,t - 5)^n$ trails.
- For $n = 5$ and $t = 10$: more than 10^{10} trails
- one ant $\longrightarrow$ one aircraft

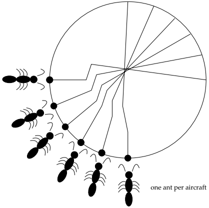




One ant per cluster or one ant per aircraft

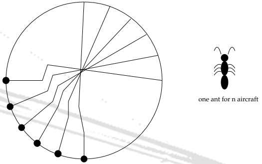
- one ant $\longrightarrow$ one cluster
- for n aircraft and t timesteps : $(12\,t - 5)^n$ trails.
- For $n = 5$ and $t = 10$: more than 10^{10} trails
- one ant $\longrightarrow$ one aircraft
- for n aircraft and t timesteps : $n\,(12\,t - 5)$ trails.

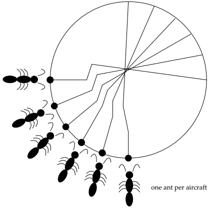




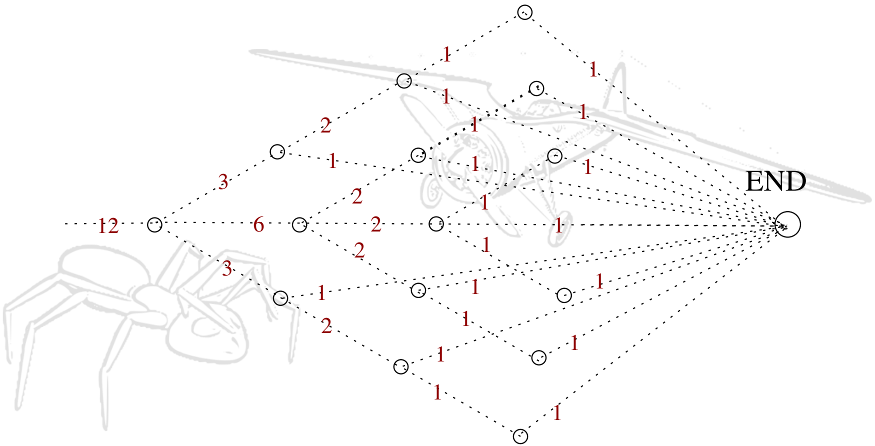
One ant per cluster or one ant per aircraft

- one ant $\rightarrow$ one cluster
- for n aircraft and t timesteps : $(12\,t-5)^n$ trails.
- For $n=5$ and $t=10$: more than 10^{10} trails
- one ant $\rightarrow$ one aircraft
- for n aircraft and t timesteps : $n\,(12\,t-5)$ trails.
- For $n=30$ and $t=20$: more than 7050 trails instead of 10^{71}





Initial amount of pheromones on the graph



Algorithm description (1)

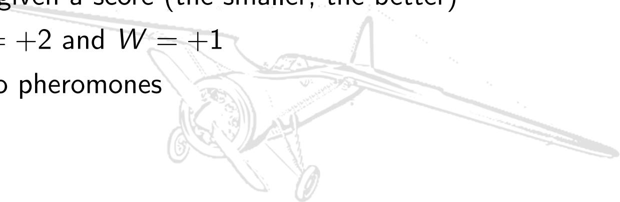
- Each path is given a score (the smaller, the better)

Algorithm description (1)

- Each path is given a score (the smaller, the better)
- $U = +0$, $V = +2$ and $W = +1$

Algorithm description (1)

- Each path is given a score (the smaller, the better)
- $U = +0$, $V = +2$ and $W = +1$
- Conflict $\rightarrow$ no pheromones



Algorithm description (1)

- Each path is given a score (the smaller, the better)
- $U = +0$, $V = +2$ and $W = +1$
- Conflict $\rightarrow$ no pheromones
- No conflict $\rightarrow$

$$\Delta\tau = \frac{n - n_{\mathrm{out}}}{n} \cdot \frac{\tau_0}{s_{\mathrm{path}}}$$

where n_{out} is the number of ”lost” ants, τ_0 the original quantity of pheromones, and s_{path} the score of the path followed by the ant.

Algorithm description (1)

- Each path is given a score (the smaller, the better)
- $U = +0$, $V = +2$ and $W = +1$
- Conflict $\rightarrow$ no pheromones
- No conflict $\rightarrow$

$$\Delta\tau = \frac{n - n_{\text{out}}}{n} \cdot \frac{\tau_0}{s_{\text{path}}}$$

  where n_{out} is the number of "lost" ants, τ_0 the original quantity of pheromones, and s_{path} the score of the path followed by the ant.
- At each node, the next edge is chosen with a probability depending on its quantity of pheromones.

Algorithm description (2)

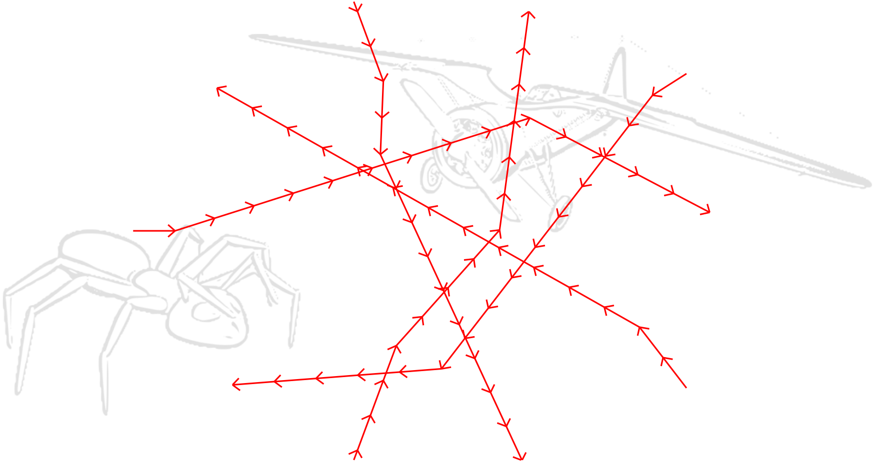
- An evaporation principle : the amount of pheromones is decreased by $x\%$ (in the examples $x = 10\%$) at the end of each iteration.

Algorithm description (2)

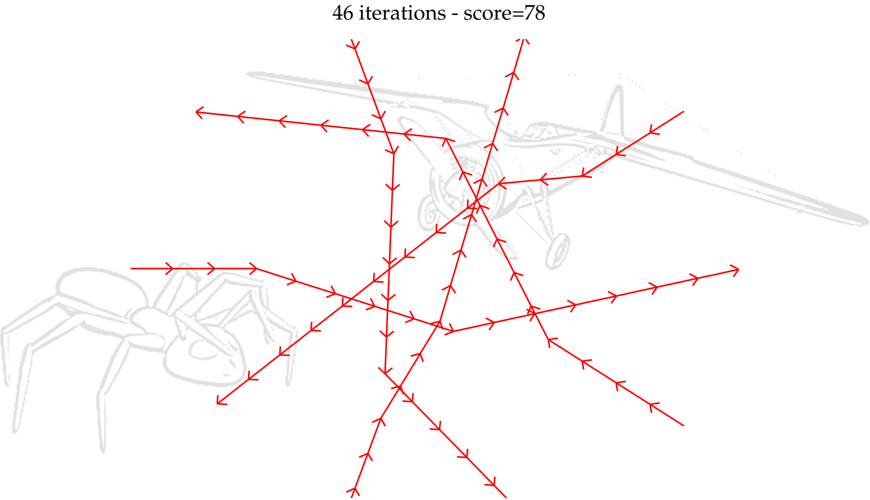
- An evaporation principle : the amount of pheromones is decreased by $x\%$ (in the examples $x = 10\%$) at the end of each iteration.
- Ending criteria : the score obtained by each bunch of ants no longer decreases.

Example of 5 aircraft conflict resolution

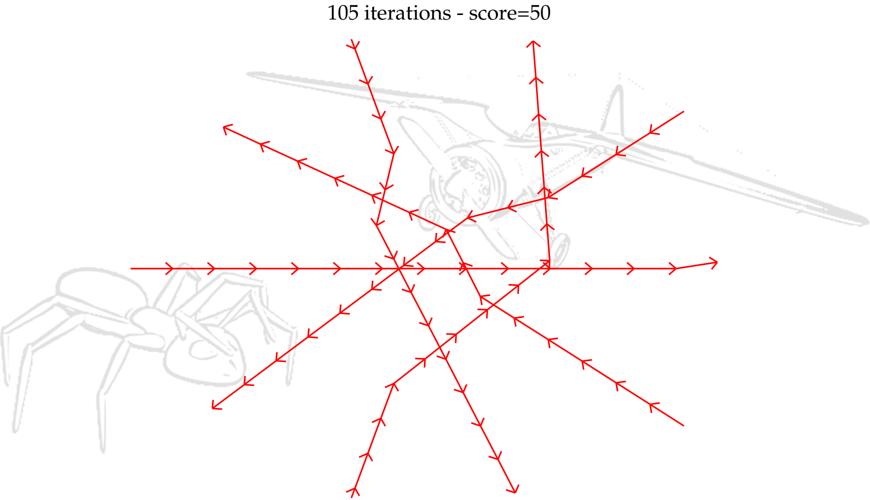
18 iterations - score=89



Example of 5 aircraft conflict resolution



Example of 5 aircraft conflict resolution



Algorithm improvement : constraint relaxation

- High density areas $\rightarrow$ no ants are able to solve every conflict

Algorithm improvement : constraint relaxation

- High density areas $\rightarrow$ no ants are able to solve every conflict
- $\Rightarrow$ Relax the conflict resolution constraint : accept ants with r remaining conflicts

Algorithm improvement : constraint relaxation

- High density areas $\rightarrow$ no ants are able to solve every conflict
- $\Rightarrow$ Relax the conflict resolution constraint : accept ants with r remaining conflicts
- When solutions are found for a certain number of ants, the constraint is reinforced

Algorithm improvement : constraint relaxation

- High density areas $\rightarrow$ no ants are able to solve every conflict
- $\Rightarrow$ Relax the conflict resolution constraint : accept ants with r remaining conflicts
- When solutions are found for a certain number of ants, the constraint is reinforced
- Example : define r as the minimum number of conflicts of the least conflicting ant

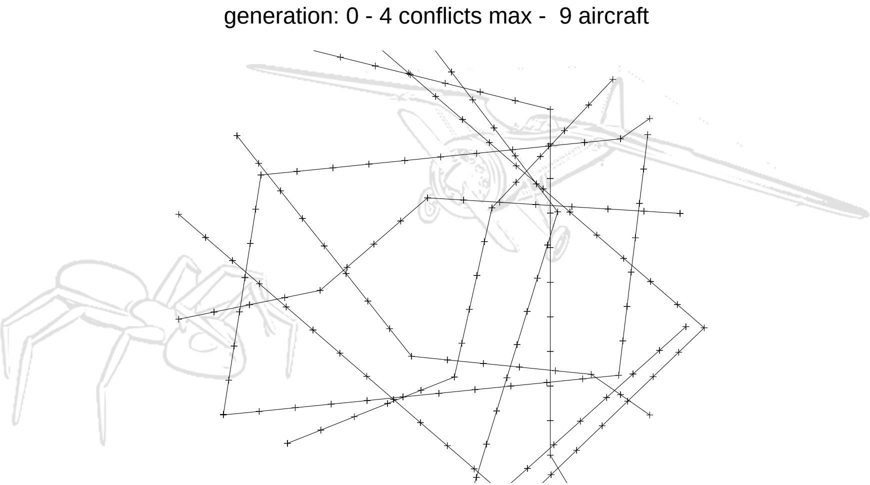
Algorithm improvement : constraint relaxation

- High density areas $\rightarrow$ no ants are able to solve every conflict
- $\Rightarrow$ Relax the conflict resolution constraint : accept ants with r remaining conflicts
- When solutions are found for a certain number of ants, the constraint is reinforced
- Example : define r as the minimum number of conflicts of the least conflicting ant
- r is the number of allowed conflicts per ant at the first generation.

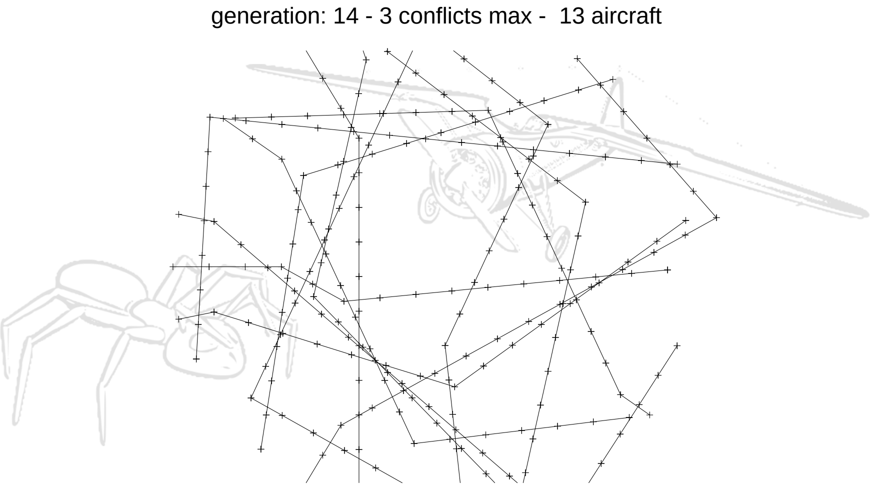
Algorithm improvement : constraint relaxation

- High density areas $\rightarrow$ no ants are able to solve every conflict
- $\Rightarrow$ Relax the conflict resolution constraint : accept ants with r remaining conflicts
- When solutions are found for a certain number of ants, the constraint is reinforced
- Example : define r as the minimum number of conflicts of the least conflicting ant
- r is the number of allowed conflicts per ant at the first generation.
- Reduce r when the number of ants having less than r conflicts is higher than $\frac{n}{r}$

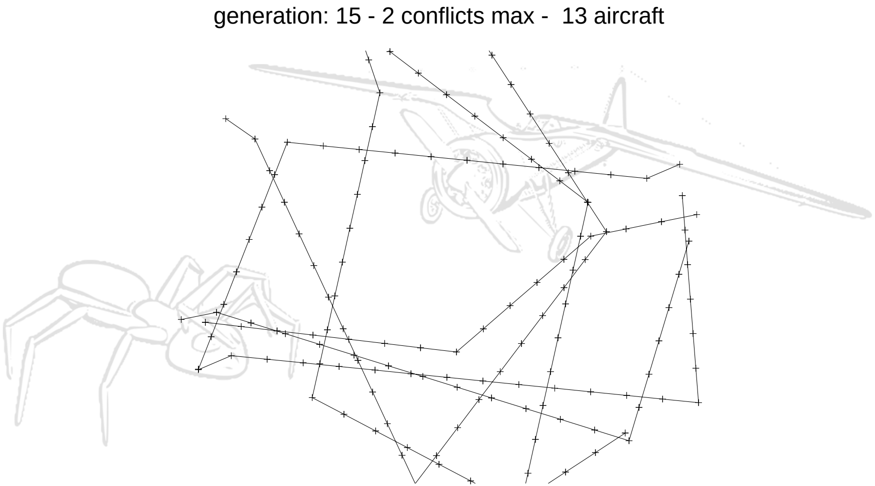
Example of 30 aircraft conflict resolution



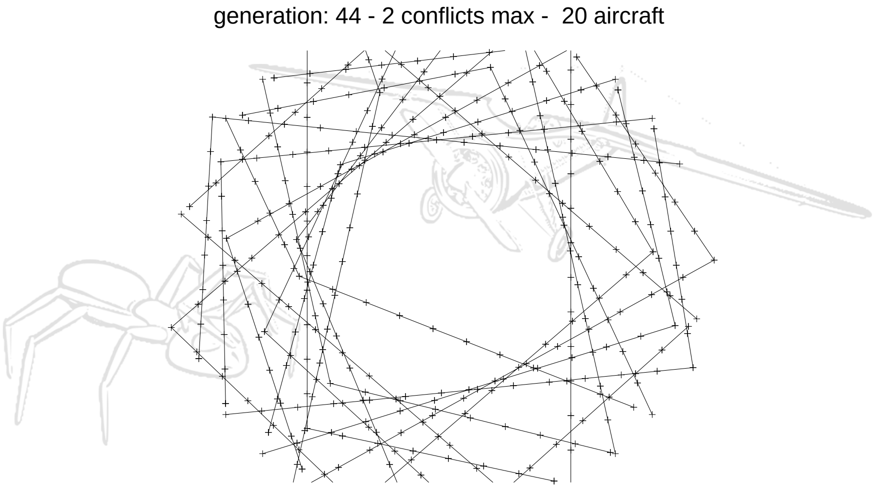
Example of 30 aircraft conflict resolution



Example of 30 aircraft conflict resolution

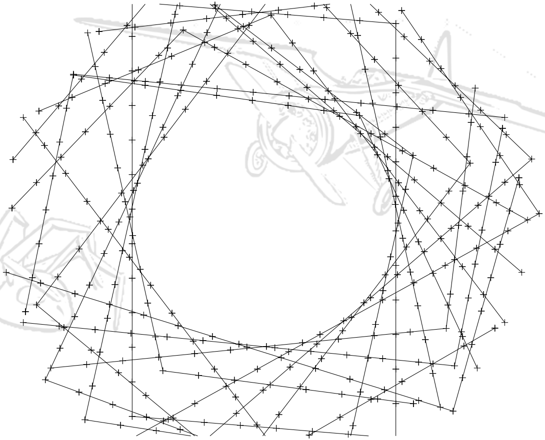


Example of 30 aircraft conflict resolution

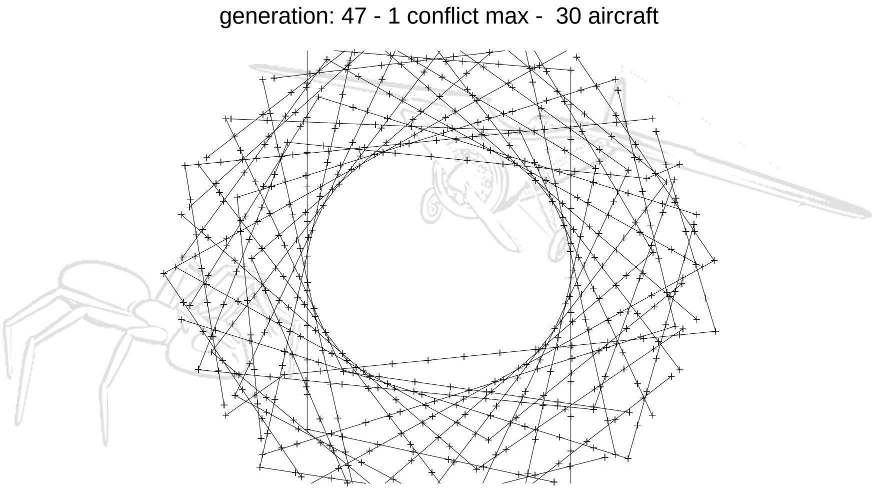


Example of 30 aircraft conflict resolution

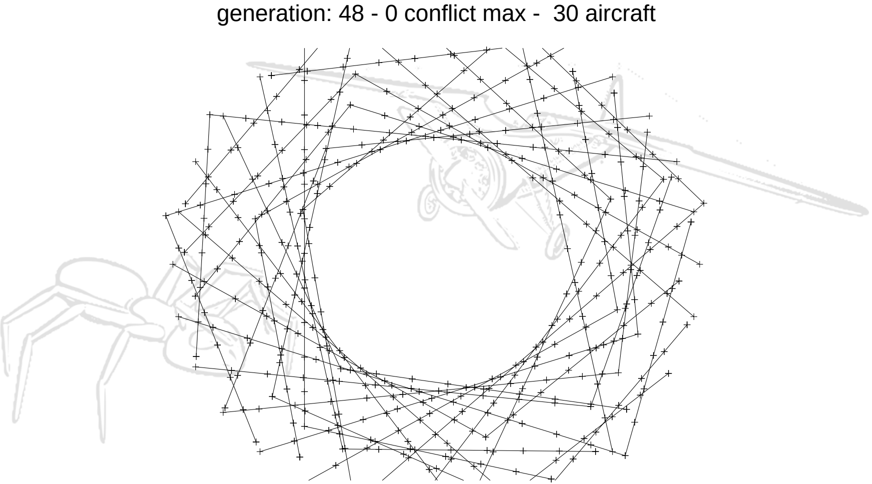
generation: 45 - 1 conflict max - 20 aircraft



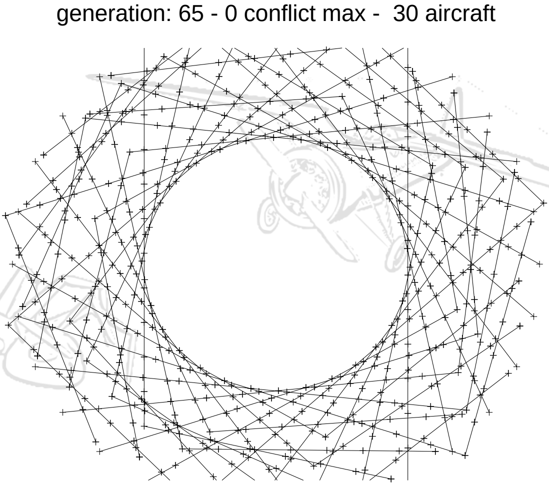
Example of 30 aircraft conflict resolution



Example of 30 aircraft conflict resolution

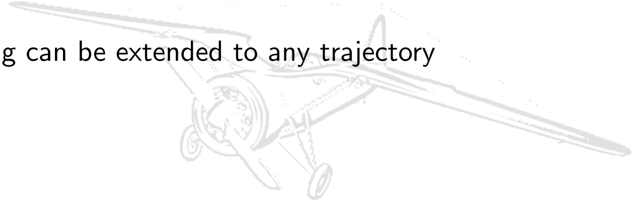


Example of 30 aircraft conflict resolution

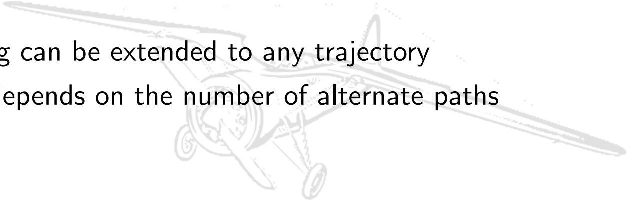


Conclusion

- The modeling can be extended to any trajectory



Conclusion



- The modeling can be extended to any trajectory
- Complexity depends on the number of alternate paths available

Conclusion

- The modeling can be extended to any trajectory
- Complexity depends on the number of alternate paths available
- Results will be compared to the existing ERCOS (using GAs)

Conclusion

- The modeling can be extended to any trajectory
- Complexity depends on the number of alternate paths available
- Results will be compared to the existing ERCOS (using GAs)
- Stochastic optimization : no guarantee of solution or optimum

Conclusion

- The modeling can be extended to any trajectory
- Complexity depends on the number of alternate paths available
- Results will be compared to the existing ERCOS (using GAs)
- Stochastic optimization : no guarantee of solution or optimum
- No effective tool offered to controllers without significant enhancement of ground TP