Optimisation globale non déterministe

Recuit simulé et
algorithmes évolutionnaires

Optimisation stochastique

- Méthodes d'optimisation qui ne requièrent pas de régularité sur les fonctions à optimiser
- Méthodes couteuses en temps de calcul qui ne garantissent pas de trouver l'optimum.
- Les résultats de convergence ne s'appliquent pas en pratique.

Le recuit simulé

- Initialisation: on part d'un point x_0 choisi au hasard dans l'espace de recherche.
  - On construit $x_{n+1}=x_n+B(0,s)$
  - On fait évoluer la température de recuit: $t_{n+1}=H(t_n)$
  - Si $f(x_{n+1})<f(x_n)$ alors on conserve x_{n+1}
  - Si $f(x_{n+1})>f(x_n)$ alors :
    - Si $|f(x_{n+1})-f(x_n)|<e^{-k\,t}$ alors on conserve x_{n+1}
    - Si $|f(x_{n+1})-f(x_n)|>e^{-k\,t}$ alors on conserve x_n

Paramètres importants

- Le schéma de recuit H détermine la façon dont l'algorithme converge.
  - Trop rapide=>L'algorithme converge vers un minimum local
  - Trop lent=>L'algorithme converge trop lentement.
- Le déplacement B(0,s) doit balayer suffisamment l'espace sans trop déplacer le point.

Efficacité

- Les algorithmes de recuit sont utiles sur des problèmes trop difficiles pour les techniques déterministes.
- On leur préfèrera des algorithmes de type génétique quand on peut construire des croisements qui ont un « sens ».

Algorithmes génétiques

- Techniques d'optimisation s'appuyant sur des techniques dérivées de la génétique et de l'évolution naturelle:
  - Reproduction
  - Croisement
  - Mutation
- Apparus aux Etats-Unis dans les années 60 à travers les travaux de John Holland
- Popularisés par David Goldberg.

Codage d'un élément et création de population

- Soit x, variable de la fonction f(x) à optimiser sur $[x_{min},x_{max}]$.
- On réécrit x : 2^n $(x-x_{min})/(x_{max}-x_{min})$
- On obtient alors un nombre compris dans l'intervalle $[0,2^n]$, soit une chaine de n bits:
  - Pour n=8: 01001110
  - Pour n=16: 0100010111010010
- On tire n éléments au hasard et les code comme ci-dessus.

Croisement

- On choisit deux parents :
  - 01100111
  - 10010111
- On tire au sort un site de croisement (3):
  - 011|00111
  - 100|10111
- On récupère les deux enfants:
  - 011|10111
  - 100|00111

Mutation

- On sélectionne un élément:
  - 01101110
- On sélectionne un site de mutation (5):
  - 01101110
- On inverse la valeur du bit:
  - 01100110

Reproduction

- Pour chaque élément x_i on calcule
  - $f(x_i)$ et $S=\Sigma(f(x_i))$
- Pour chaque x_i on calcule
  - $p(x_i)=f(x_i)/\Sigma(f(x_i))$
- On retire les n éléments de la population k+1 à partir des n éléments de la population k en prenant comme probabilité de tirage $p(x_i)$

Exemple de reproduction

- Soit f(x)=4x(1-x)
- x prend ses valeurs dans [0,1[

Séquence	Valeur	$U(x)$	% de chance de reproduction	% cumulés	Après reproduction
10111010	0.7265625	0.794678	0.794678 / 2.595947 = 0.31	0.31	11011110
11011110	0.8671875	0.460693	0.460693 / 2.595947 = 0.18	0.31+0.18=0.49	10111010
00011010	0.1015625	0.364990	0.364990 / 2.595947 = 0.14	0.49+0.14=0.63	01101100
01101100	0.4218750	0.975586	0.975586 / 2.595947 = 0.37	0.62+0.37=1.00	01101100
=		2.595947			

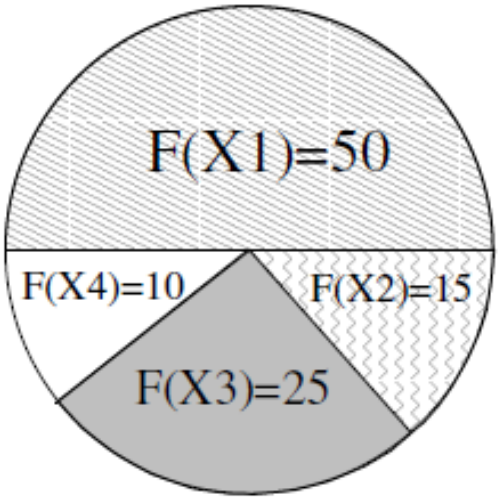
Fonctionnement d'un AG

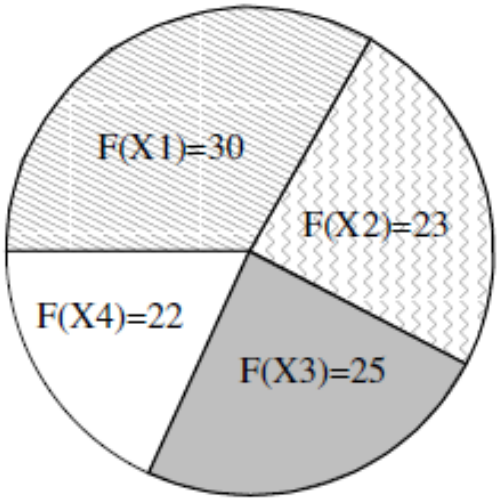
- Etape 1: reproduction
- Etape 2: croisement
- Etape 3: mutation
- Etape 4: test de fin, et retour à l'étape 1.

Le scaling

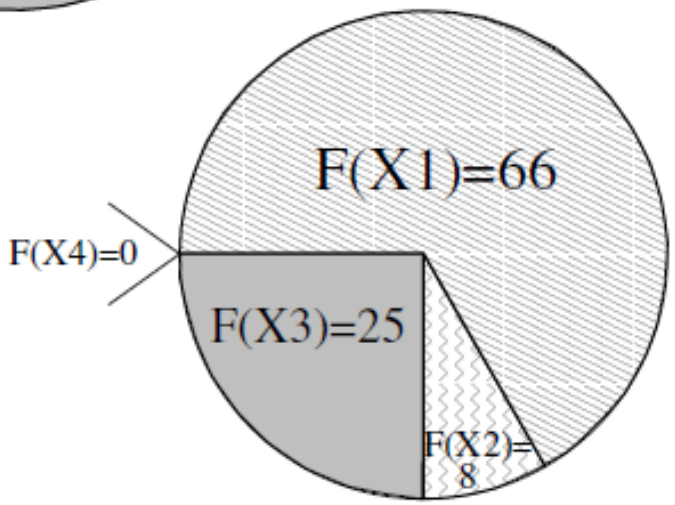
- Le fonctionnement de l'algorithme dépend fortement de la valeur de l'adaptation.
- Au lieu d'utiliser directement f(x) comme adaptation, on la « met à l'échelle » en appliquant une fonction croissante.
- Exemples:
  - 5 (f(x)-10)/3: augmente la pression
  - 0.2 f + 20 : diminue la pression

Exemple de scaling





0.2 F + 20



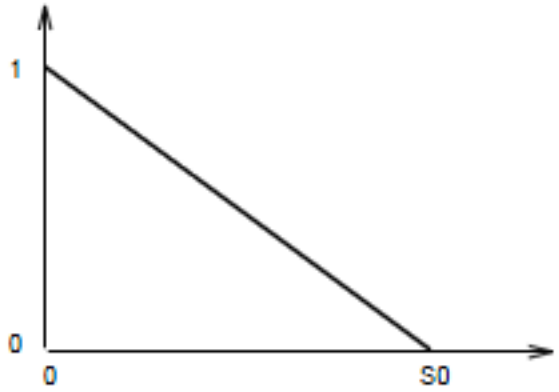
5/3 (F – 10)

Le sharing

- La pression de sélection peut entrainer une convergence locale trop rapide.
- Le sharing modifie l'adaptation en fonction du nombre d'éléments voisins de l'élément courant:
  - $f_s(x_i) = f(x_i) / \Sigma_j \, s(d(x_i, x_j))$
  - s est une fonction décroissante.
  - $d(x_i, x_j)$ mesure la distance entre i et j

Le sharing

- Le sharing demande la mise en place d'une fonction distance sur l'espace des variables.
- Forme générale de s:



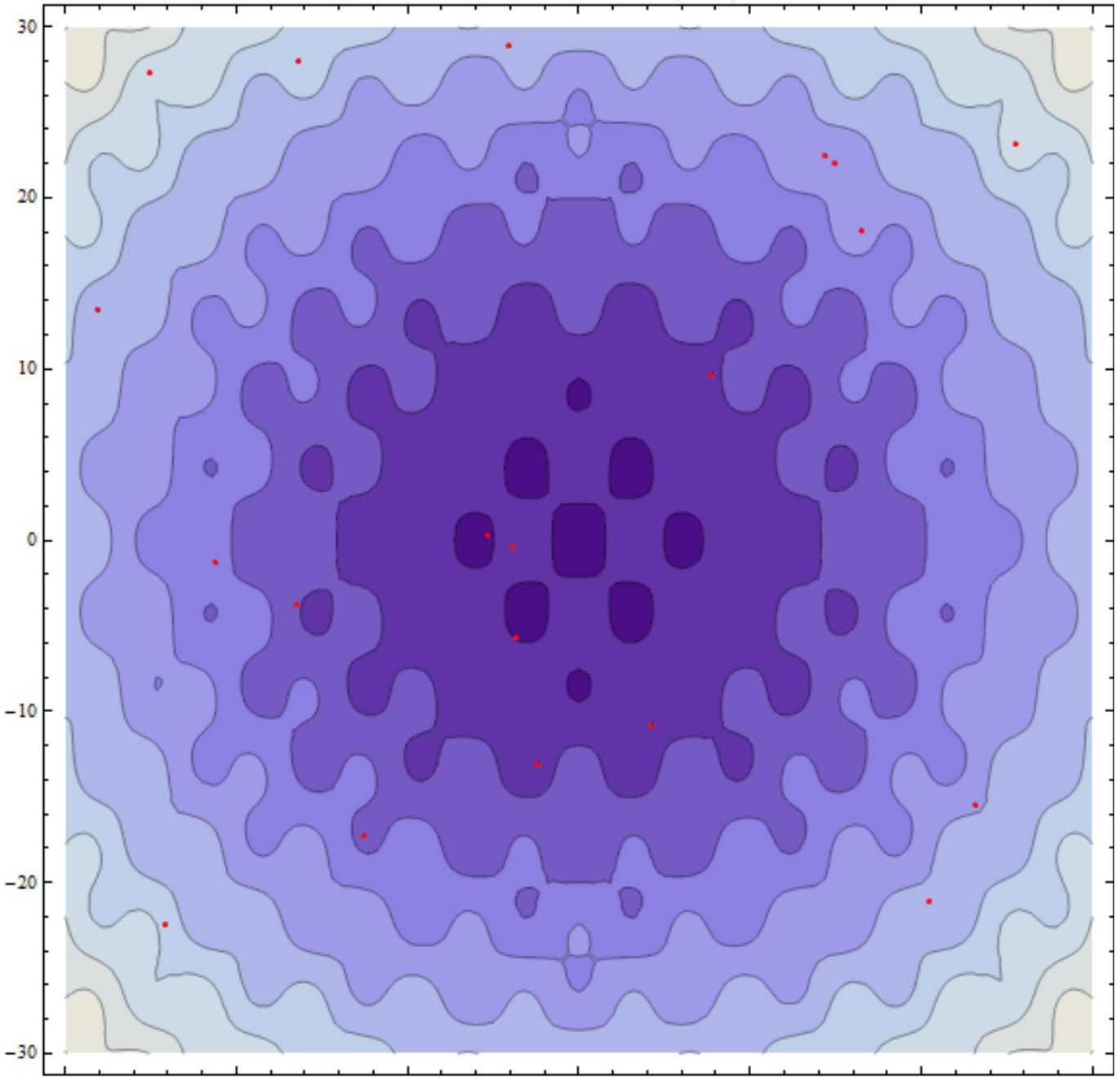
Problème du codage en chaine de bit

- Deux éléments très différents au niveau du génotype peuvent avoir des phénotypes identiques.
  - Sur un codage simple de [0,1] en 8 bits:
  - 10000000 et 01111111 représentent quasiment la même valeur (1/2) mais leur distance de Hamming est maximale.
  - On peut utiliser des codes de Grey, ou employer des représentations adaptées.

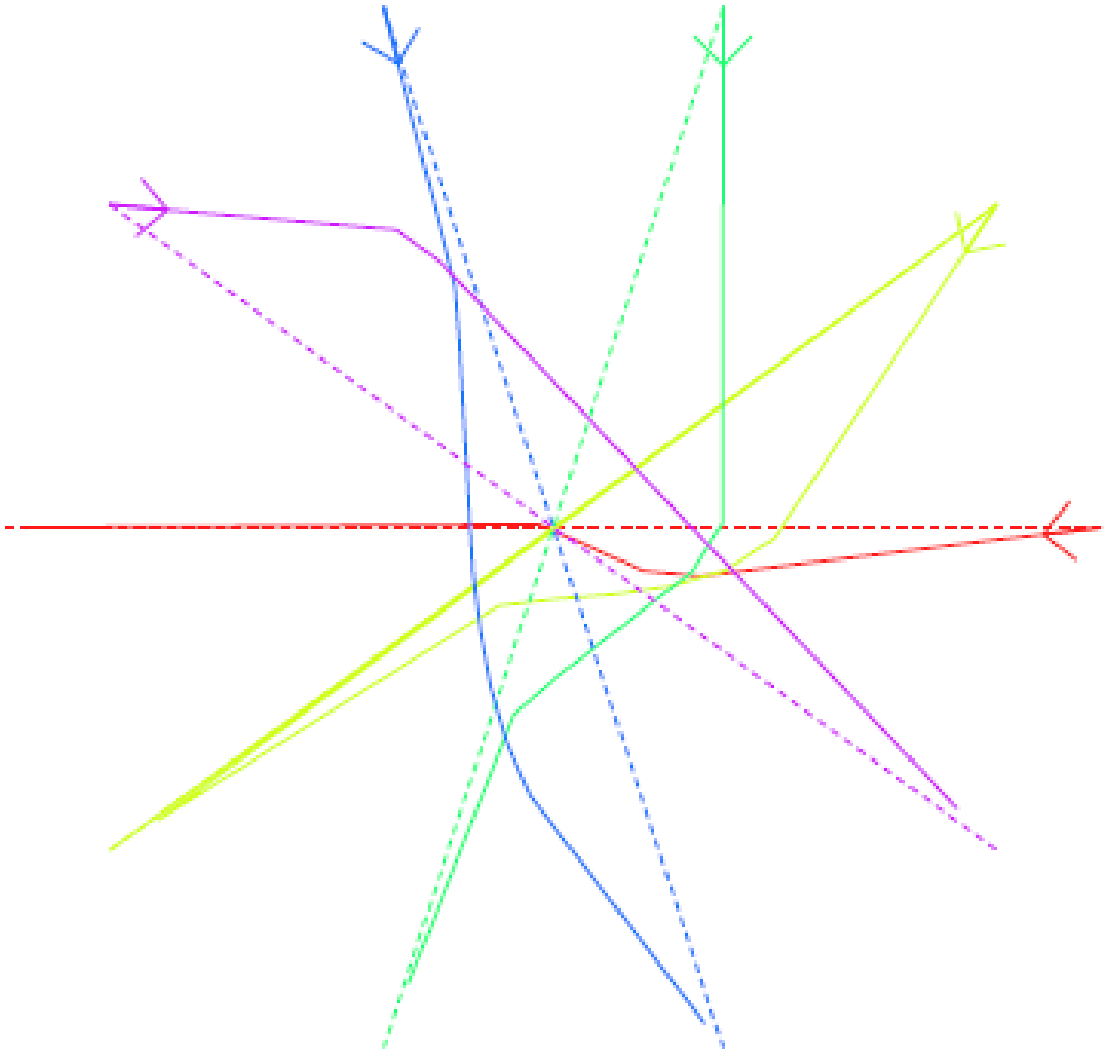
Représentation adaptée

- Pour les fonctions à variable réelle, on code directement la variable par sa valeur
- Croisement:
  - $y_1 = \alpha\, x_1 + (1-\alpha)\, x_2$
  - $y_2 = (1-\alpha)\, x_1 + \alpha\, x_2$
  - α pris dans [0.5,1.5]
- Mutation:
  - $y_1 = x_1 + B(0,\sigma)$

$$\frac{1}{100}\left(x^2+y^2\right)-\cos(x)\cos\left(\frac{y}{\sqrt{2}}\right)$$



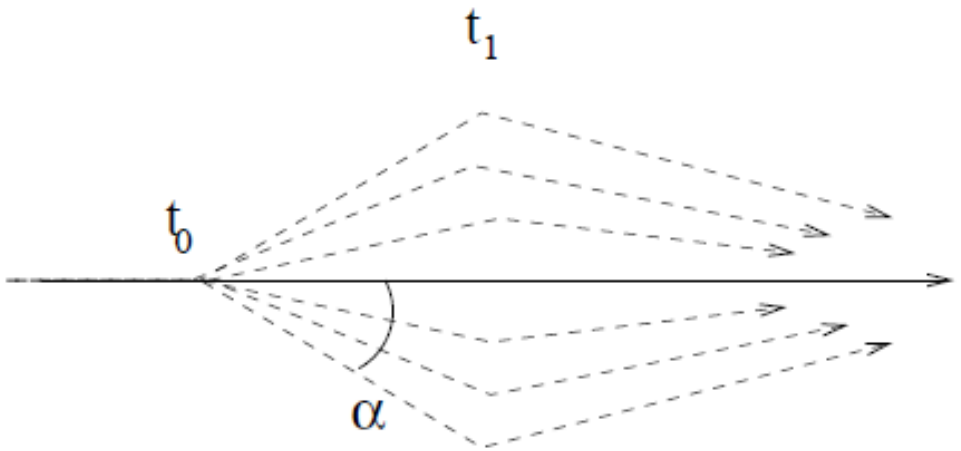
Résolution de conflits aériens



Modélisation

Modélisation

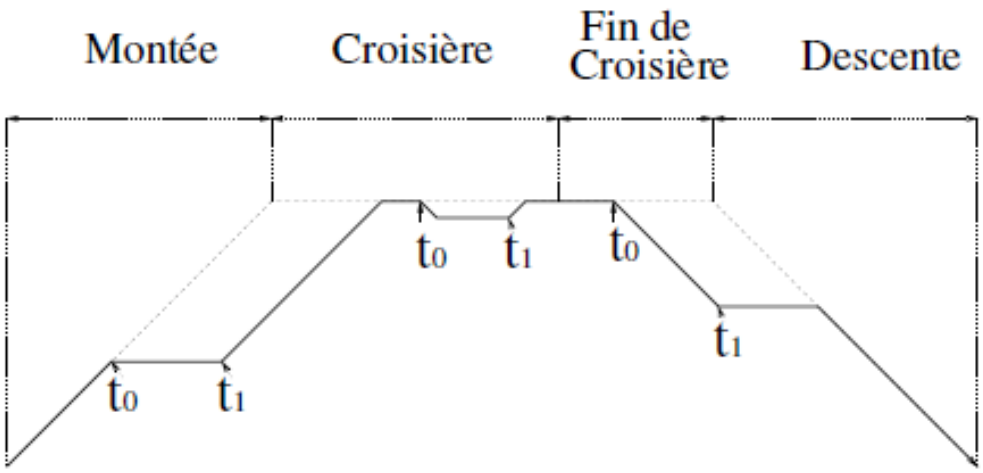
- Une seule manœuvre par avion
- Manoeuvres horizontales
- Manoeuvres verticales
- Modélisation de l'incertitude
- $3n$ variables



Modélisation

Modélisation

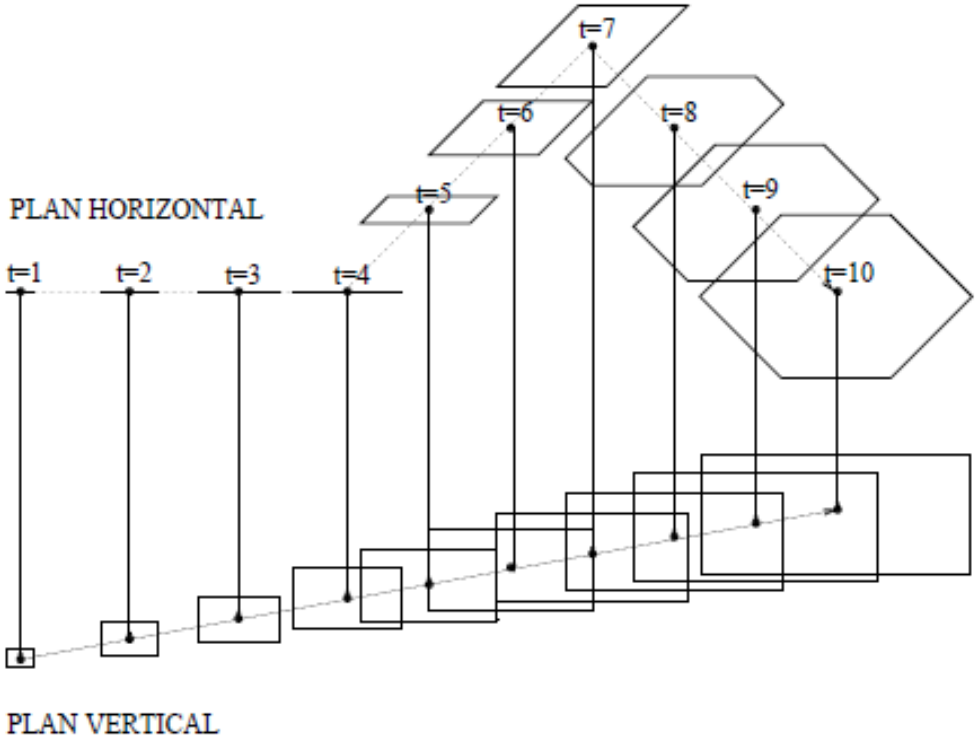
- Une seule manœuvre par avion
- Manoeuvres horizontales
- Manoeuvres verticales
- Modélisation de l'incertitude
- $3n$ variables



Modélisation

Modélisation

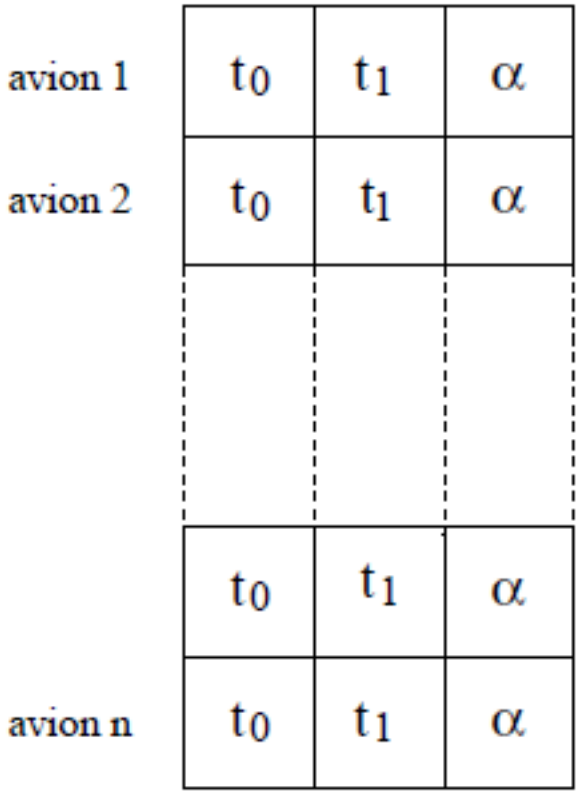
- Une seule manœuvre par avion
- Manoeuvres horizontales
- Manoeuvres verticales
- Modélisation de l'incertitude
- $3n$ variables



Modélisation

Modélisation

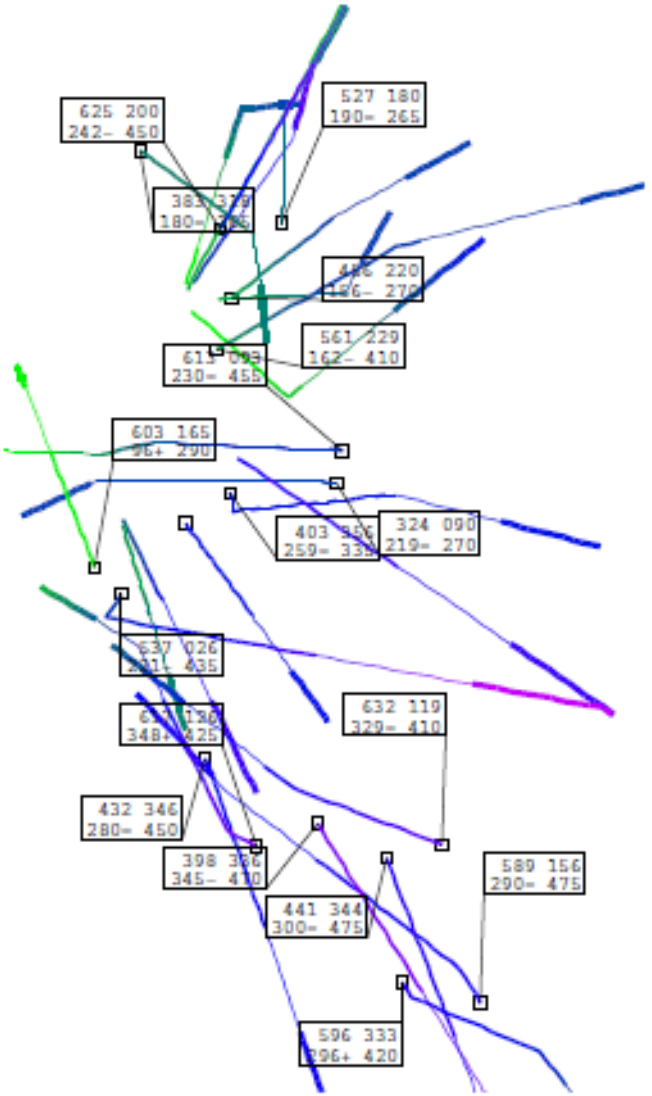
- Une seule manœuvre par avion
- Manoeuvres horizontales
- Manoeuvres verticales
- Modélisation de l'incertitude
- $3n$ variables

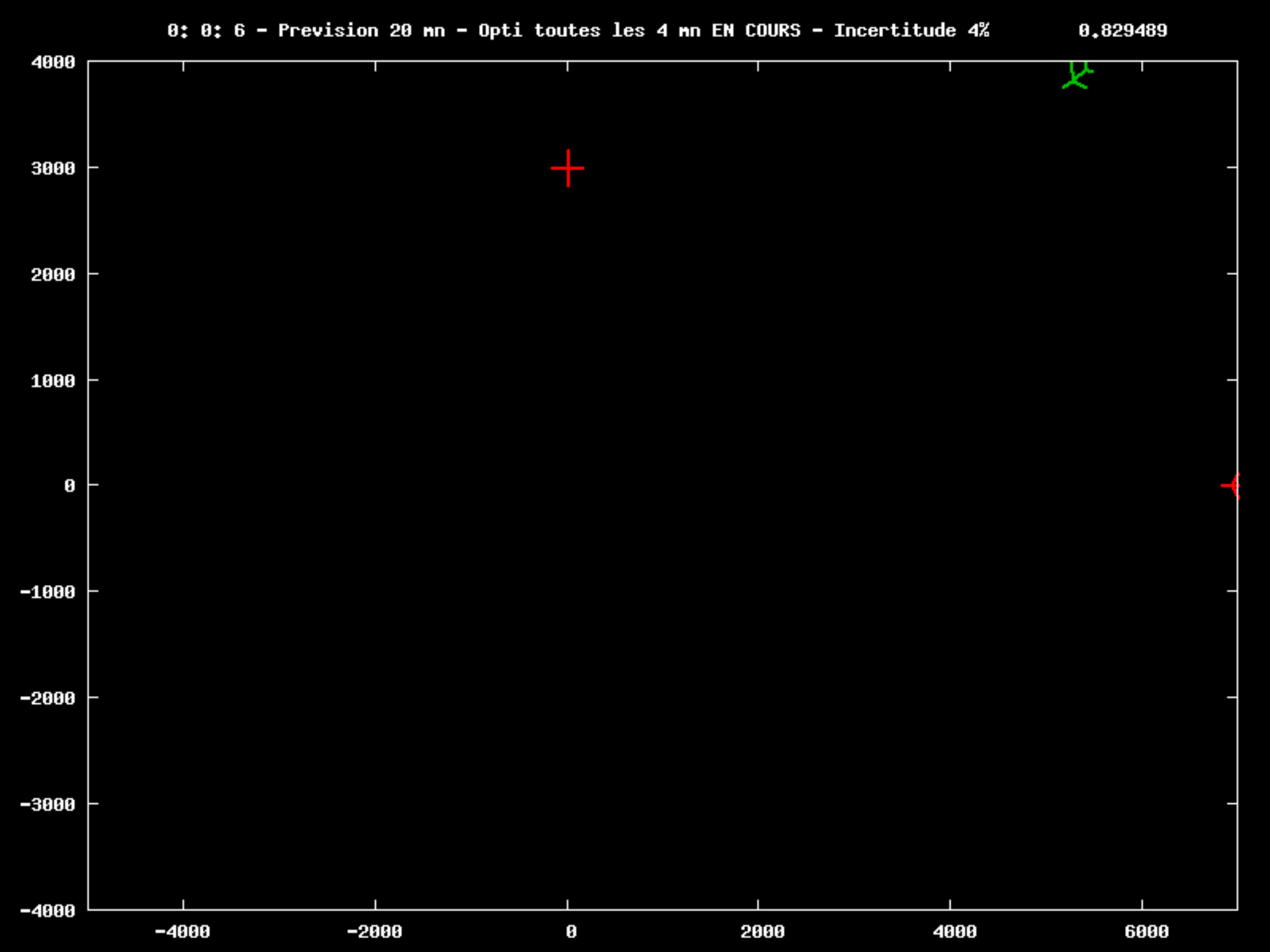


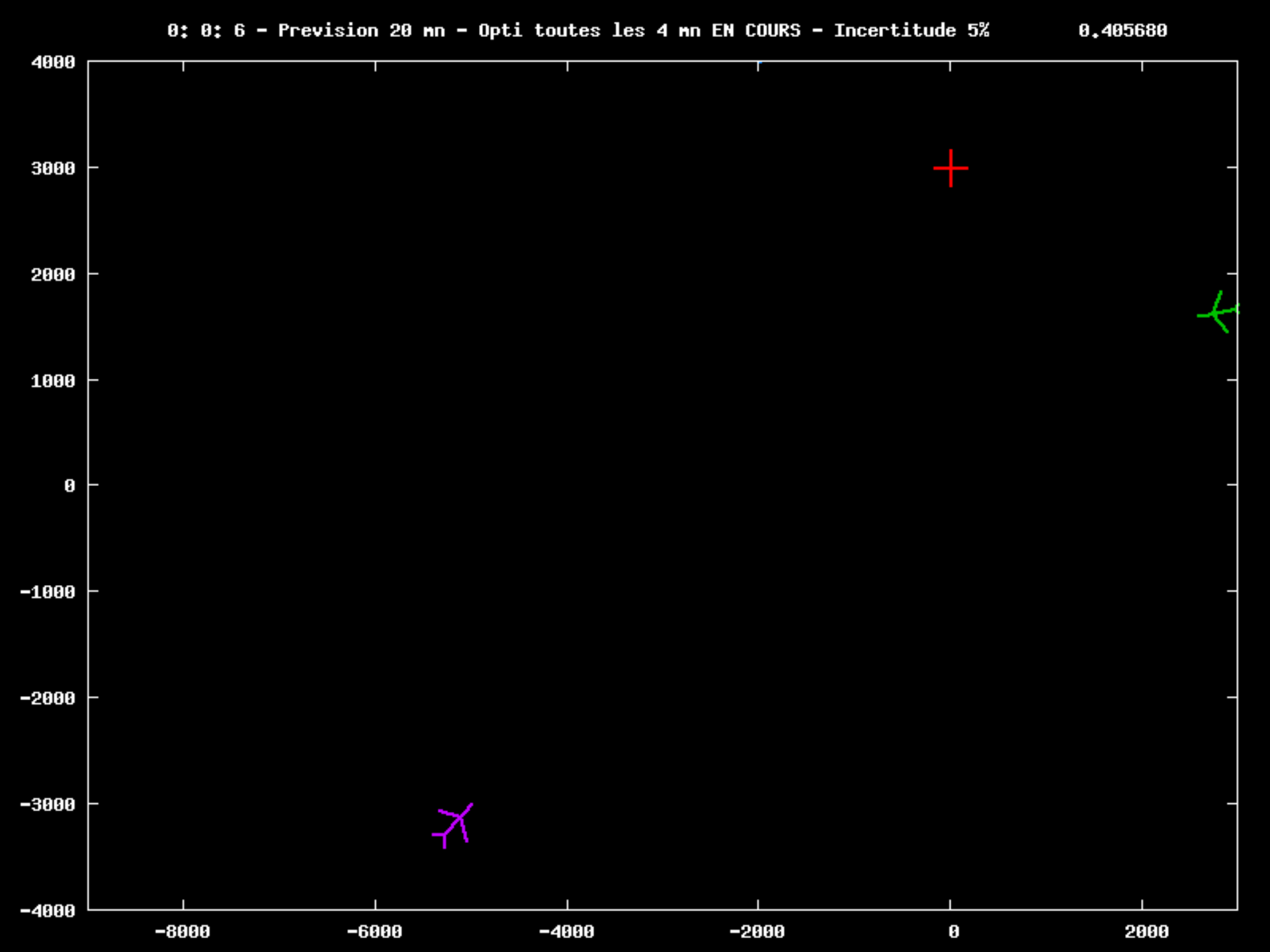
Résultats

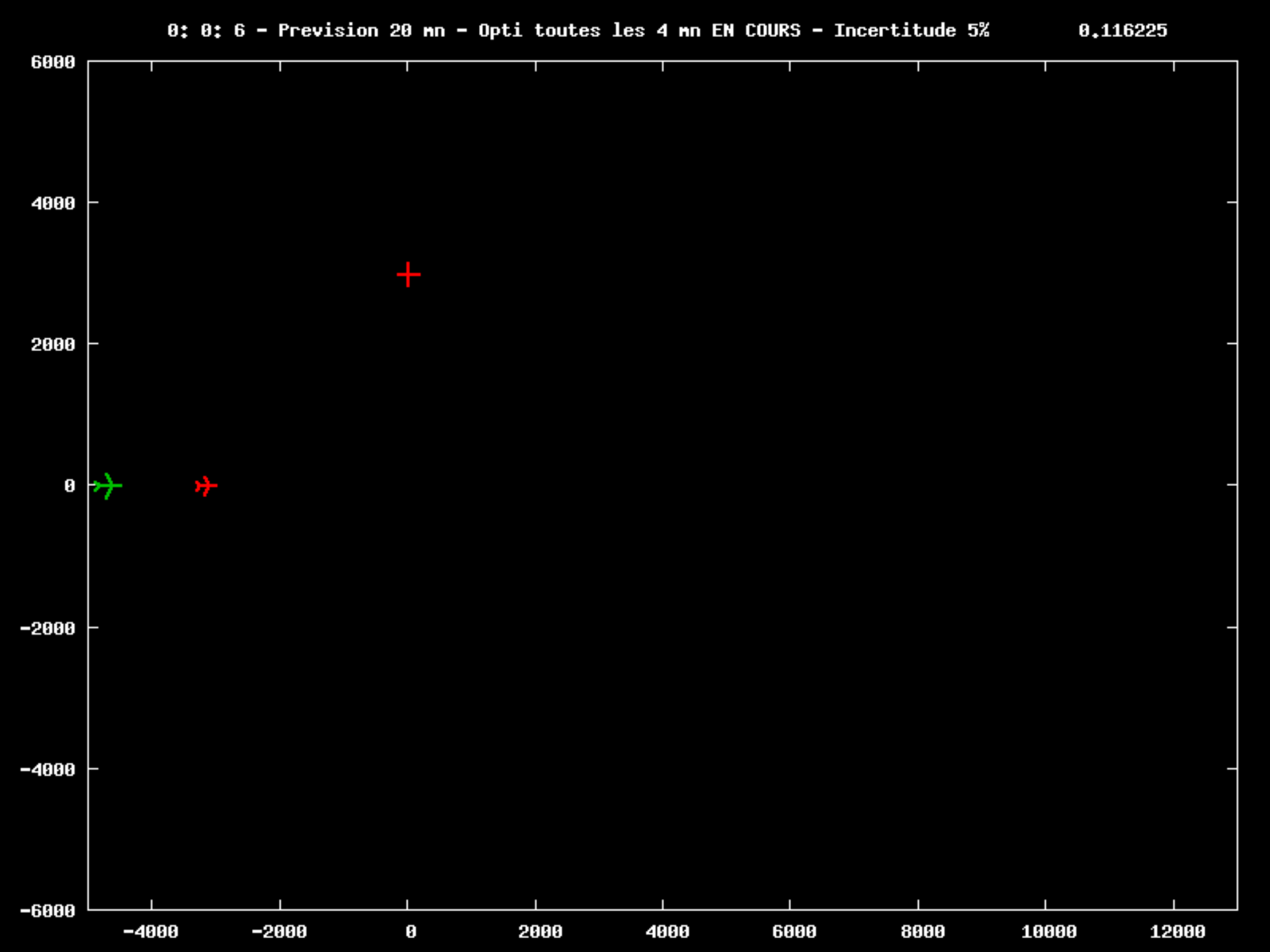
Résultats

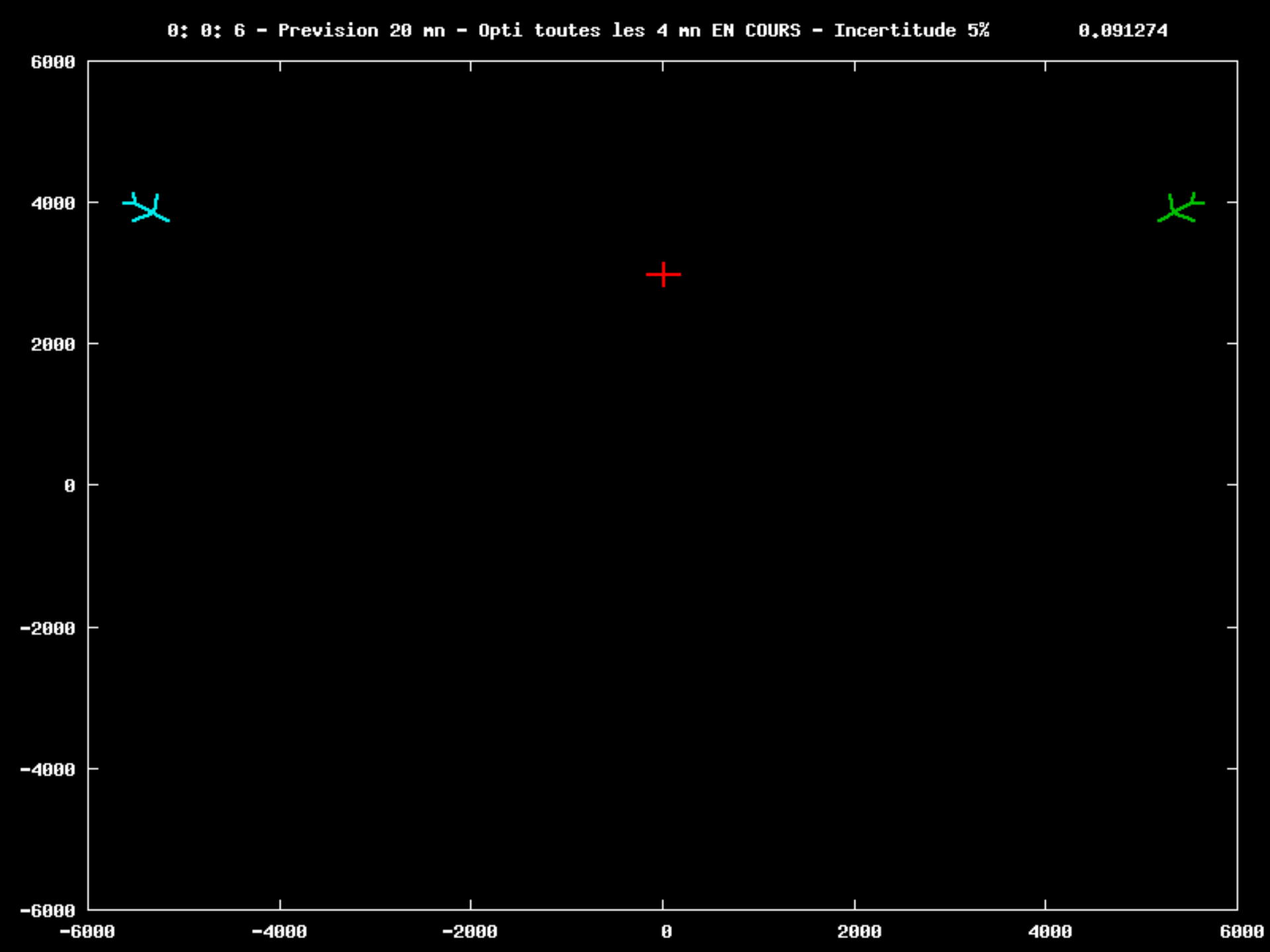
- Résout des gros conflits (30 avions)
- Intégration dans un outil de simulation (CATS/OPAS)
- Testé sur des journées de trafic réel
- Peu de restrictions sur la modélisation
- Pas de garantie d'optimalité

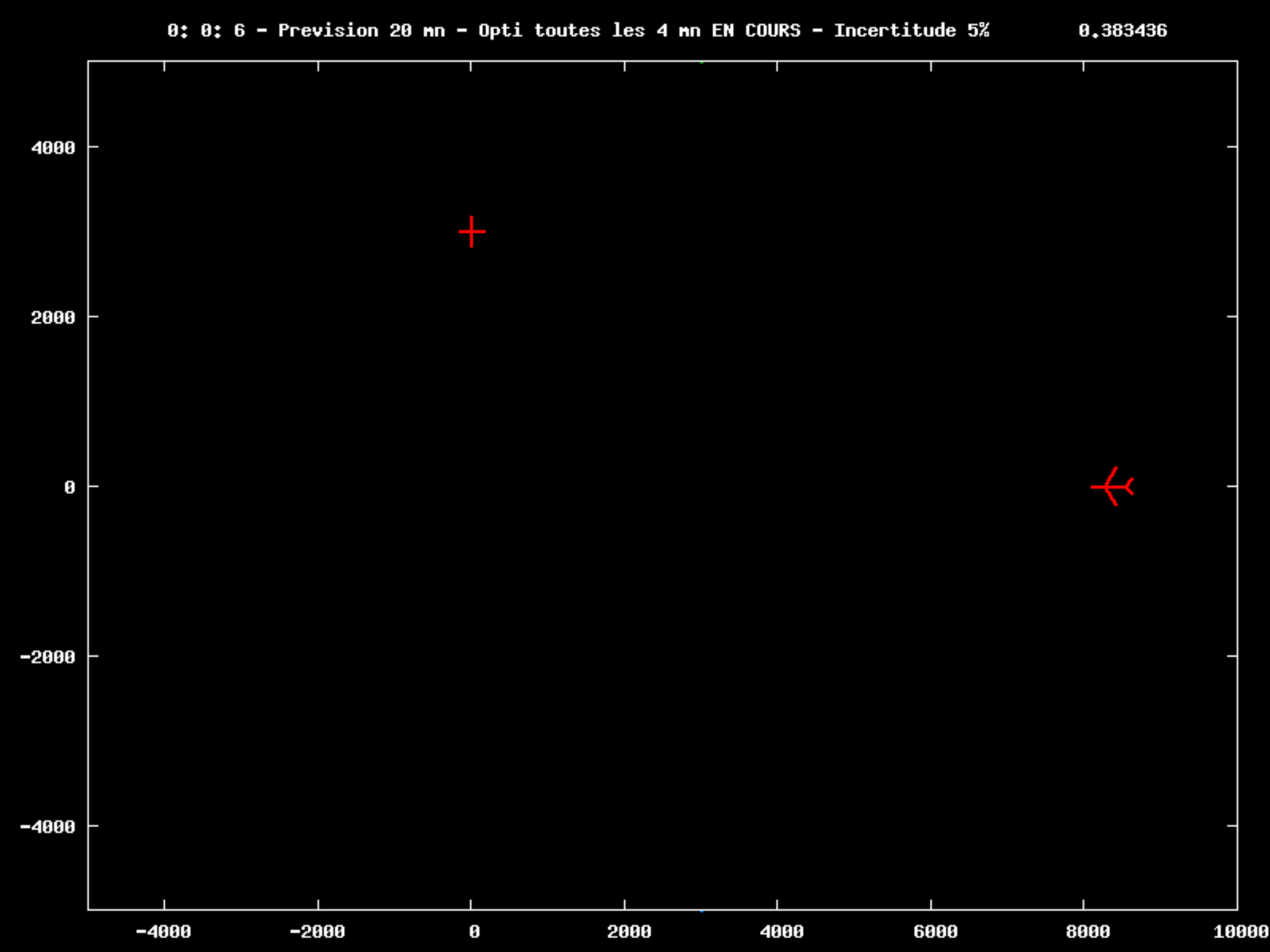


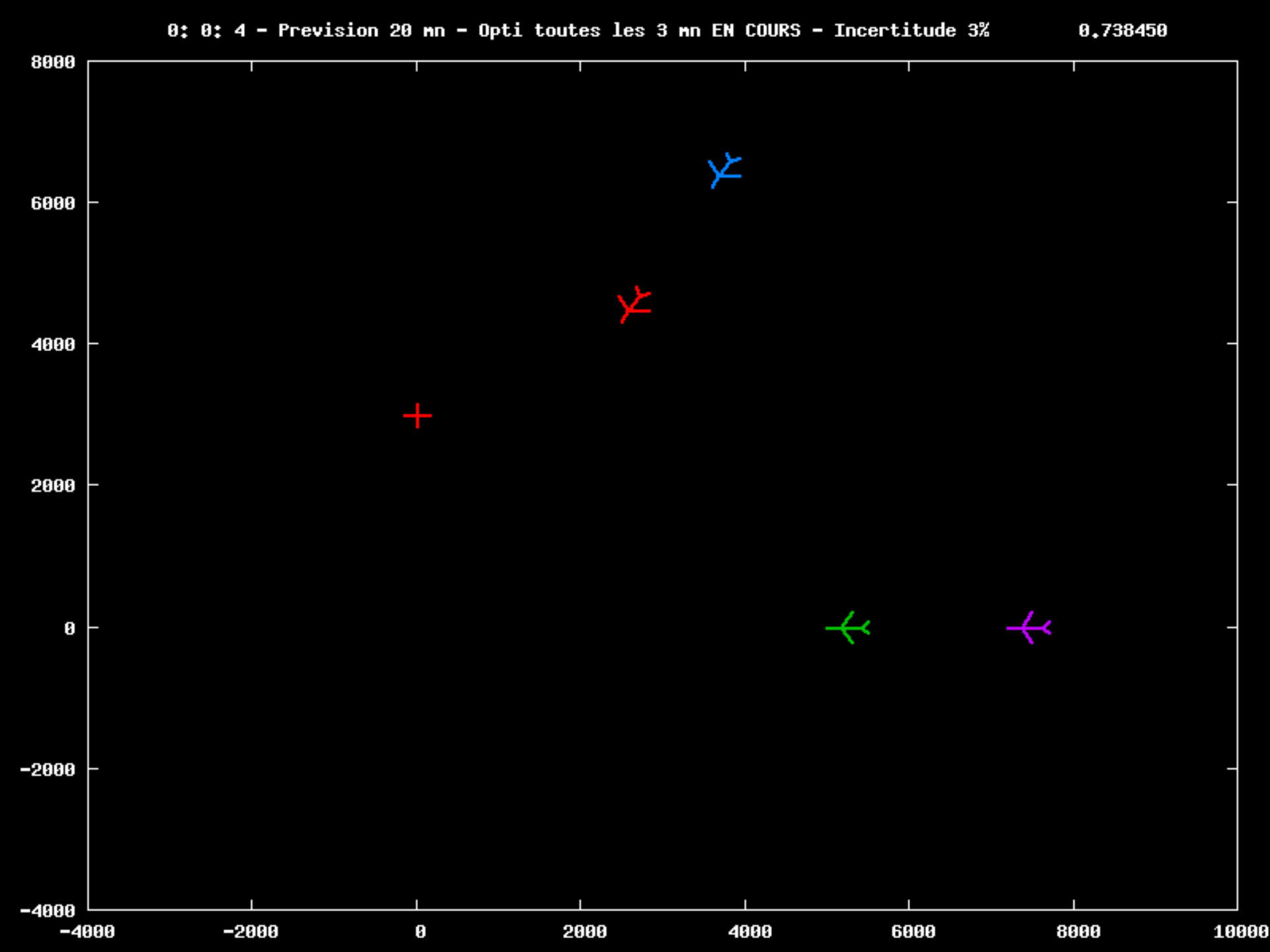

0: 0: 6 - Prevision 20 mn - Opti toutes les 4 mn EN COURS - Incertitude 4%
0.829489
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-4000
-2000
0
2000
4000
6000


0: 0: 6 - Prevision 20 mn - Opti toutes les 4 mn EN COURS - Incertitude 5%
0.405680
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-8000
-6000
-4000
-2000
0
2000

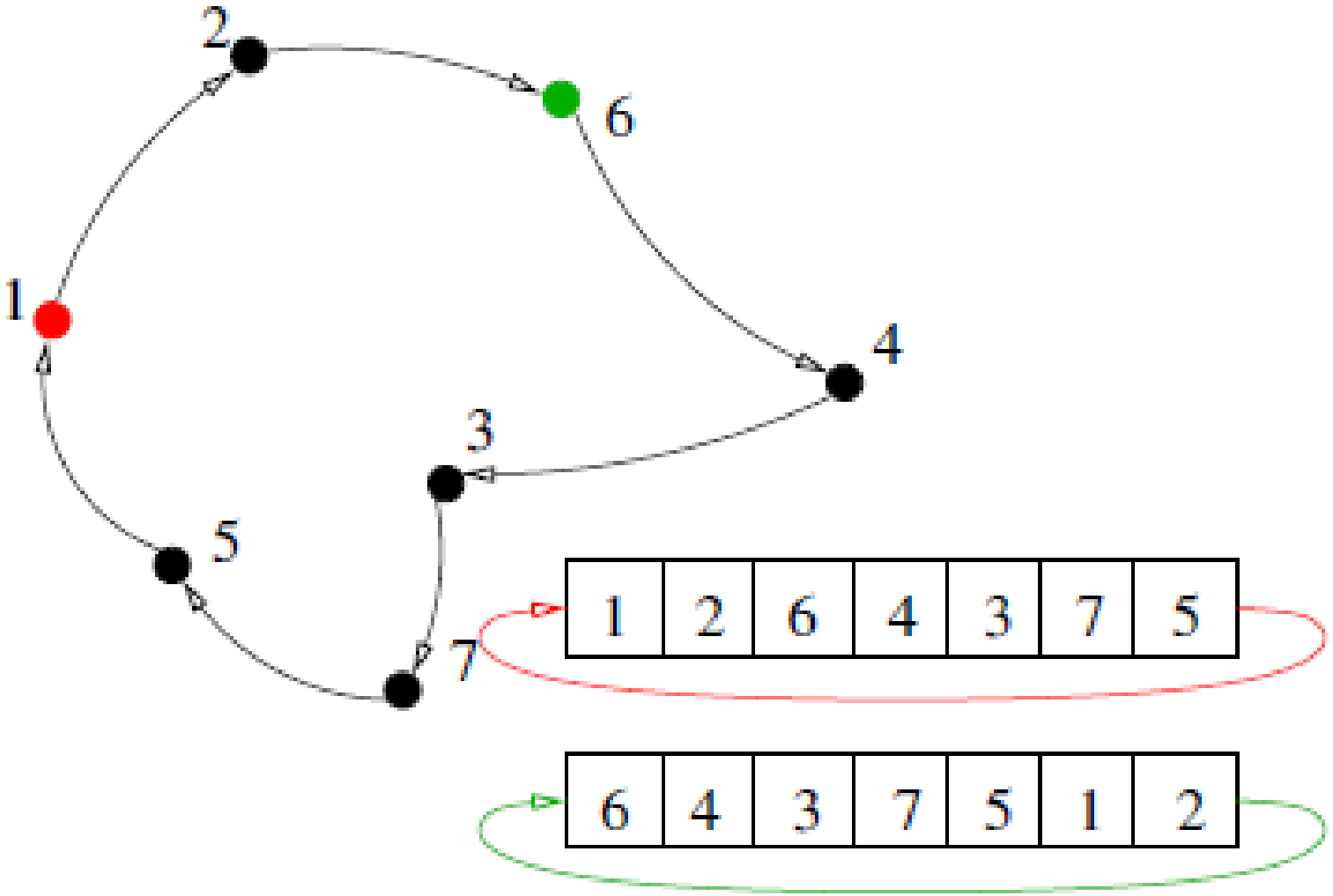

0: 0: 6 - Prevision 20 mn - Opti toutes les 4 mn EN COURS - Incertitude 5%
0.116225
6000
4000
2000
0
-2000
-4000
-6000
-4000
-2000
0
2000
4000
6000
8000
10000
12000


0: 0: 6 - Prevision 20 mn - Opti toutes les 4 mn EN COURS - Incertitude 5%
0.091274
6000
4000
2000
0
-2000
-4000
-6000
-6000
-4000
-2000
0
2000
4000
6000

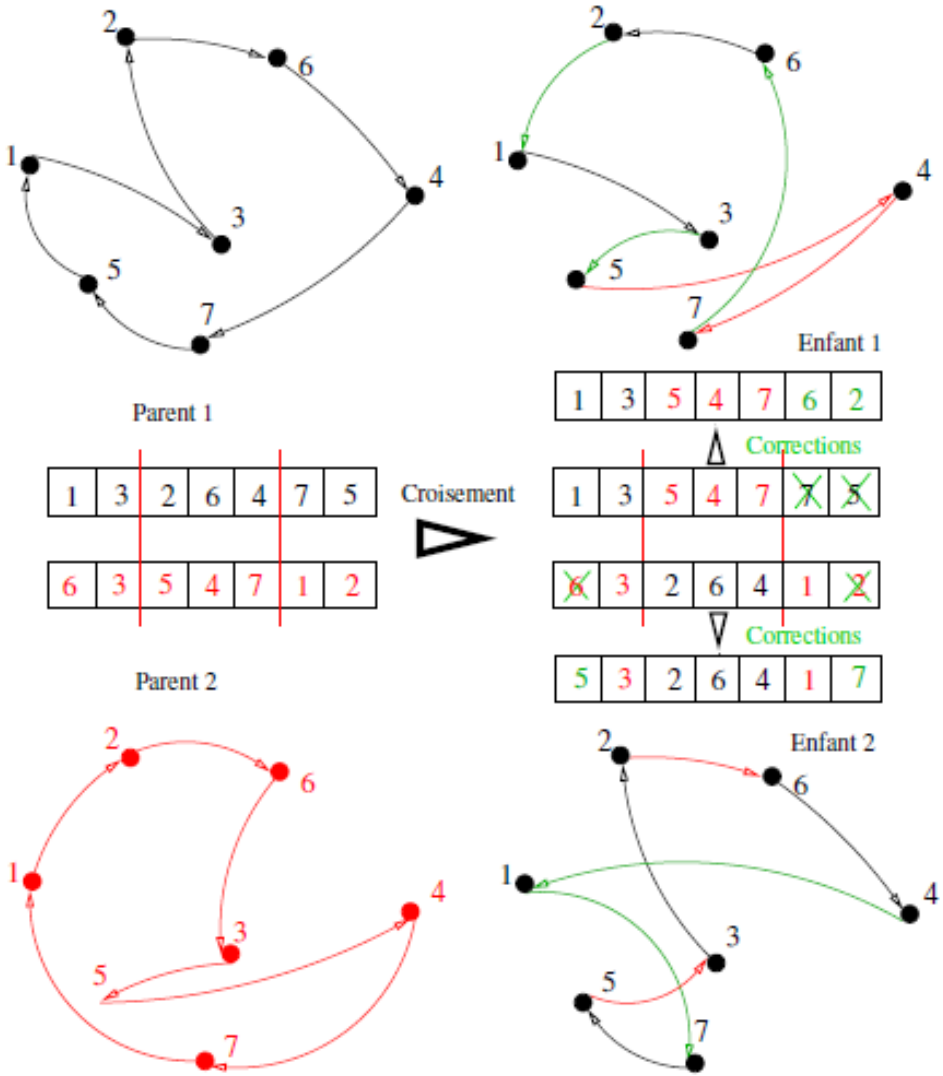

0: 0: 6 - Prevision 20 mn - Opti toutes les 4 mn EN COURS - Incertitude 5%
0.383436
4000
2000
0
-2000
-4000
-4000
-2000
0
2000
4000
6000
8000
10000


0: 0: 4 - Prevision 20 mn - Opti toutes les 3 mn EN COURS - Incertitude 3%
0.738450
8000
6000
4000
2000
0
-2000
-4000
-4000
-2000
0
2000
4000
6000
8000
10000

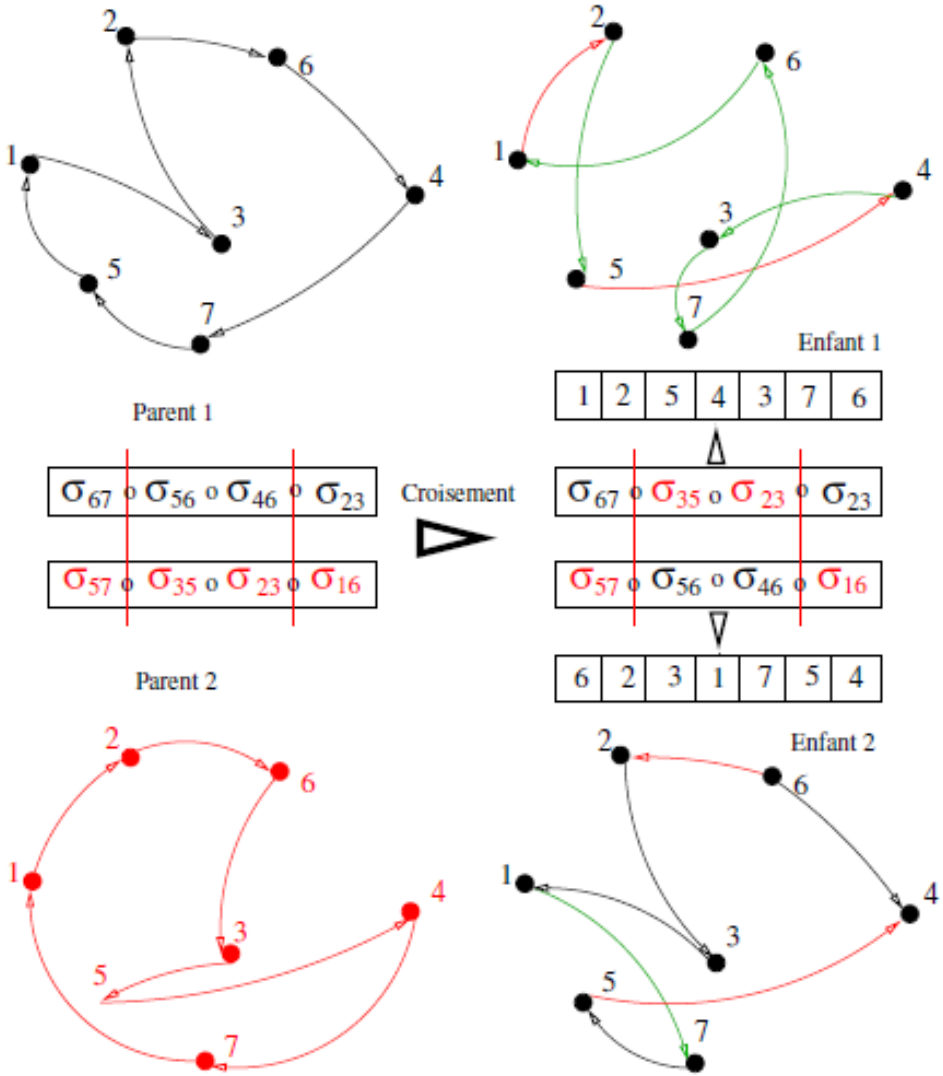
Le voyageur de commerce



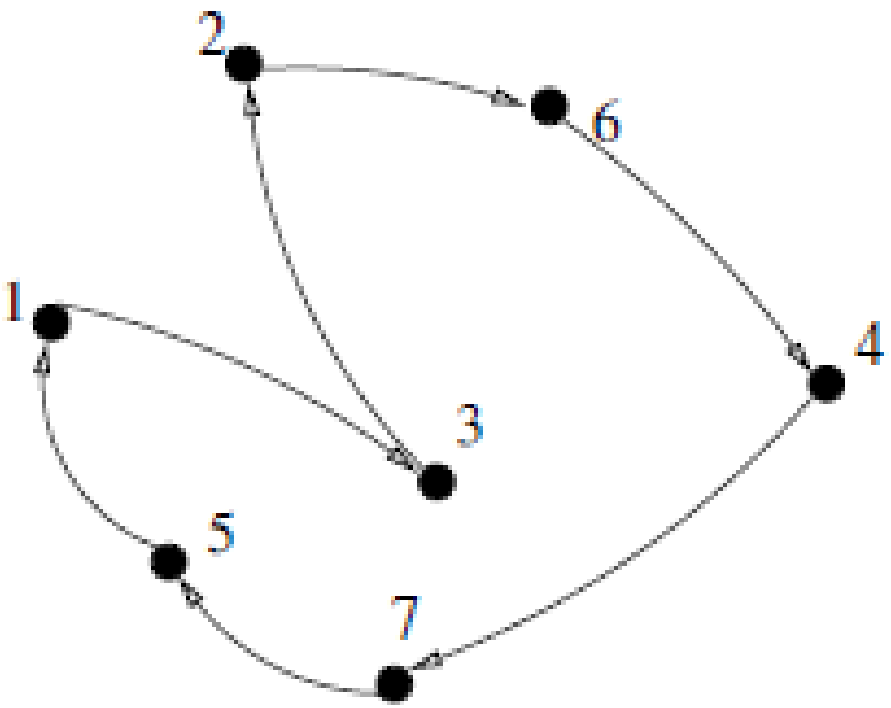
Le voyageur de commerce: croisement



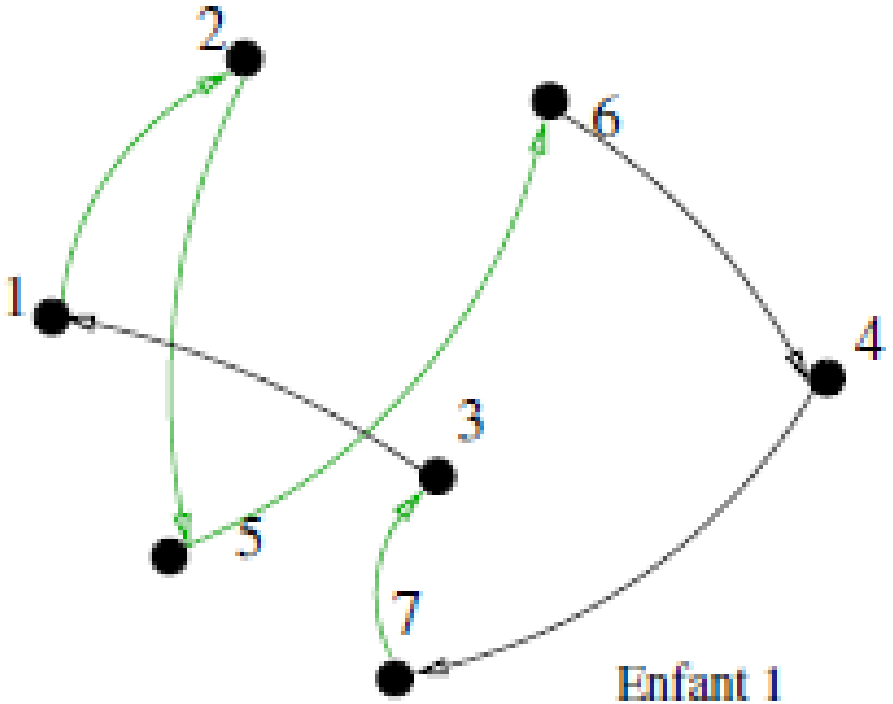
Nouveau croisement



Voyageur de commerce: mutation



Parent 1



Enfant 1

1	2	5	4	3	7	6

Mutation

$\sigma_{67} \circ \sigma_{56} \circ \sigma_{46} \circ \sigma_{23}$ → $\sigma_{67} \circ \sigma_{56} \circ \sigma_{46} \circ \sigma_{35}$