

Un système multi-agent adaptatif pour concevoir des produits complexes

Jean-Baptiste Welcomme

EADS-IW

5 mai 2008

Plan

1 Introduction

- Contexte de la conception avant-projet avion
- Objectifs de MASCODE

2 MASCODE

- Régulation des contraintes
- Comprendre le comportement global
- Optimisation

3 Résultats

- Avantages propres à MASCODE
- Comparaison sur un cas simple

4 Travaux futurs

MASCODE : Multi-Agent Systems and Multidisciplinary Aircraft Simulation for CONceptual DEsign

Plan

1 Introduction

- Contexte de la conception avant-projet avion
- Objectifs de MASCODE

2 MASCODE

- Régulation des contraintes
- Comprendre le comportement global
- Optimisation

3 Résultats

- Avantages propres à MASCODE
- Comparaison sur un cas simple

4 Travaux futurs

Introduction

Au niveau métier, la conception nécessite

- La collaboration de **métiers différents** (structures, aérodynamique, moteurs, systèmes...)
- La collaboration pour **l'intégration de composants** (ailes, fuselage, systèmes)
- La décomposition pour atteindre des **étapes +/- détaillées**

Introduction

Au niveau métier, la conception nécessite

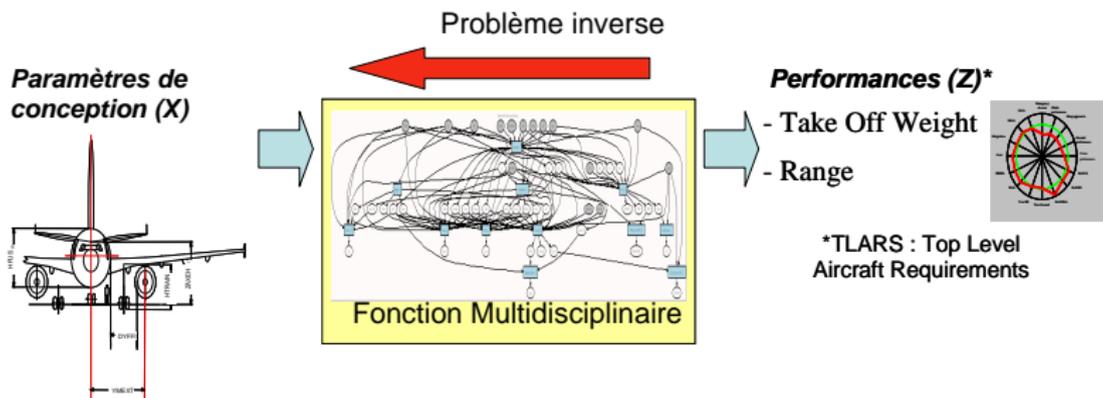
- La collaboration de **métiers différents** (structures, aérodynamique, moteurs, systèmes...)
- La collaboration pour **l'intégration de composants** (ailes, fuselage, systèmes)
- La décomposition pour atteindre des **étapes +/- détaillées**

Introduction

Au niveau métier, la conception nécessite

- La collaboration de **métiers différents** (structures, aérodynamique, moteurs, systèmes...)
- La collaboration pour **l'intégration de composants** (ailes, fuselage, systèmes)
- La décomposition pour atteindre des **étapes +/- détaillées**

Un problème inverse et interdépendant

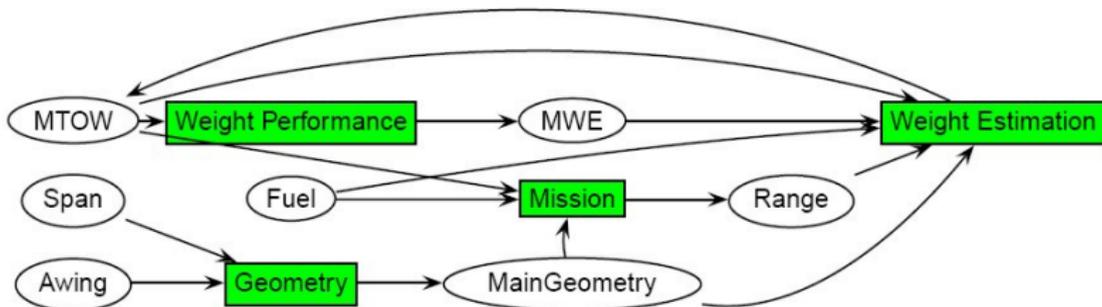


Quelques caractéristiques générales

- Nombreux paramètres (de conception et de performance)
- Des modèles disciplinaires interdépendants

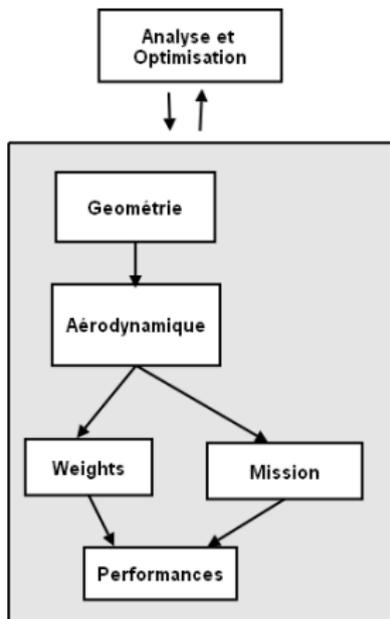
⇒ un problème inverse qui nécessite de nombreux compromis (itérations)

Un exemple d'interdépendance



- *Range* dépendant de *Fuel* et de *MaxTakeOffWeight*.
- *MaxTakeOffWeight* dépend de *Fuel* et *MaxWeightEmpty*.
- Augmenter *Range* \Rightarrow Augmenter *Fuel* \Rightarrow Augmentation de la masse structurelle \Rightarrow Augmenter *MTOW* \Rightarrow Augmenter la consommation \Rightarrow Diminuer *Range*.

Multi-Objective Optimisation (MOO)



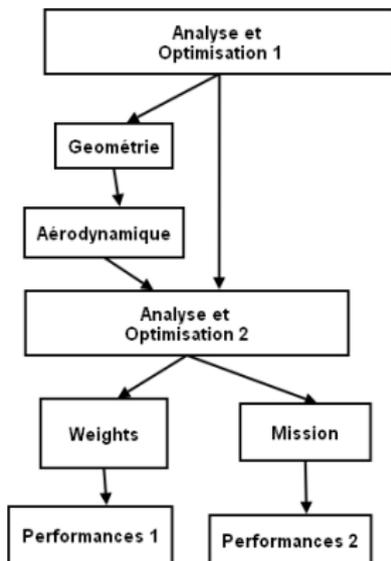
Principes

- Le problème = "boîte noire"
 - Algorithmes génétiques [Badufle, 2007]
 - Algorithmes de gradient, FSQP [Zhou, 1997]

Limites

- Pas/peu de visibilité sur les compromis
- Chaque modification \Rightarrow une exécution complète

Multi-Disciplinary Optimisation (MDO)



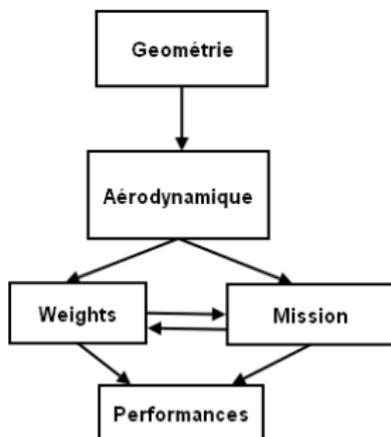
Principes

- Décomposition en plusieurs sous-problèmes [Allison, 2004]
 - L'intégration sous-pbs par optimisation

Limites

- La décomposition influence la résolution
- Les compromis sont invisibles au delà du découpage [Alexandrov, 2004]

Objectifs de MASCODE



Une modélisation

- Donner de l'autonomie aux disciplines
- Se rapprocher de l'organisation réelle
- Utiliser la structure du problème, Identifier les interdépendances, Améliorer la compréhension

Une technologie agent [Ferber, 1995]

- Systèmes distribués et adaptatifs
- Supportant les interactions d'entités autonomes

Un système multi-agent [Ferber, 1995]

Définition

Un agent est une entité autonome (logiciel) située dans un environnement et interagissant avec d'autres agents.

Concevoir un système multi-agent

- Définir les agents
- Concevoir les agents :
 - des modèles de **connaissances locales et environnementales**
 - des capacités de **communication et d'adaptation**.
- Le **comportement global du système** est issu des **interactions**
 - La cohérence du collectif est obtenue par des **négociations**

Un système multi-agent [Ferber, 1995]

Définition

Un agent est une entité autonome (logiciel) située dans un environnement et interagissant avec d'autres agents.

Concevoir un système multi-agent

- Définir les agents
- Concevoir les agents :
 - des modèles de **connaissances locales et environnementales**
 - des capacités de **communication et d'adaptation**.
- Le **comportement global du système** est issu des **interactions**
 - La cohérence du collectif est obtenue par **des négociations**

Les systèmes multi-agents adaptatifs - AMAS

Théorie des AMAS

- **Principe** : tous les agents sont coopératifs [Capera, 2004]
 - Chaque agent **améliore la situation** qu'il perçoit comme étant **la plus critique**.
 - Objectif : éviter les **situations non-coopératives**
- **Méthode de conception** [Bernon, 2003] :
 - On identifie les situations non-coopératives de l'agent (**point de vue local**)
 - On définit **des actions** pour **sortir de ces situations**
- **Résolution** : L'agent utilise ces actions et adapte son comportement pour trouver des solutions

Plan

1 Introduction

- Contexte de la conception avant-projet avion
- Objectifs de MASCODE

2 MASCODE

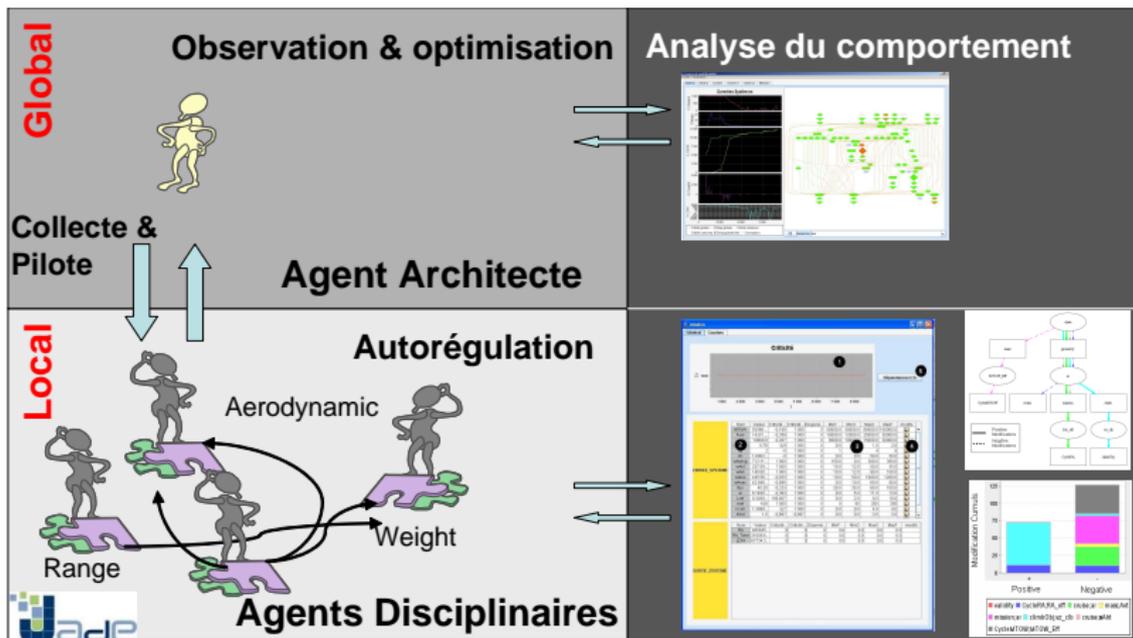
- Régulation des contraintes
- Comprendre le comportement global
- Optimisation

3 Résultats

- Avantages propres à MASCODE
- Comparaison sur un cas simple

4 Travaux futurs

Articulation des sous-parties

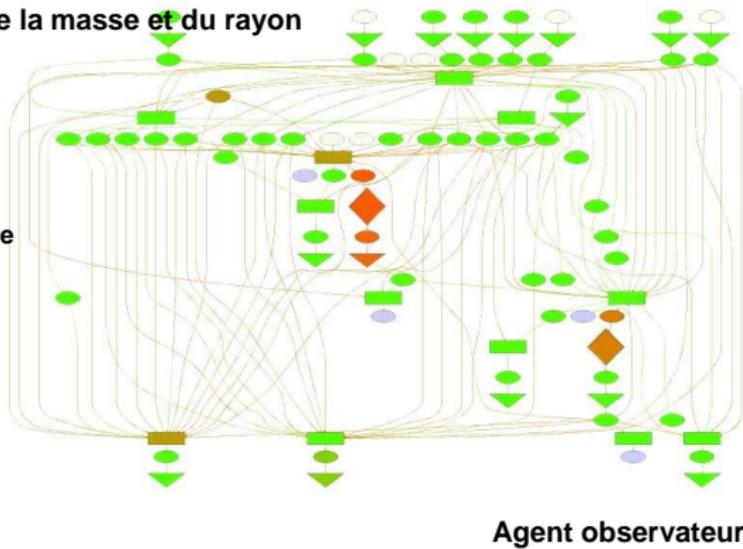
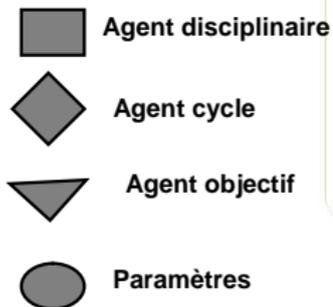


- Autorégulation / Observation et comportement global / Optimisation

Le problème et son agentification

Caractéristique du cas test :

- 8 objectifs d'entrée.
- 5 objectifs de sortie.
- Des objectives intermédiaires
- 10 modèles disciplinaires.
- 2 cycles (estimation de la masse et du rayon action)



MASCODE : Compétences de l'agent

■ Capacité de calcul (modèle disciplinaire)

- Sens direct \Rightarrow exécution
- Sens indirect \Rightarrow résolution

■ Capacité de communication

- Sens direct : « forwardMessage » 

- Nouvelle valeur exécutée

- Sens indirect : « backwardMessage » 

- Sens de la modification
- Criticité de la demande
- Origine de la demande

■ Objectifs individuels

MASCODE : Compétences de l'agent

■ Capacité de calcul (modèle disciplinaire)

- Sens direct \Rightarrow exécution
- Sens indirect \Rightarrow résolution

■ Capacité de communication

- Sens direct : « forwardMessage » 

- Nouvelle valeur exécutée

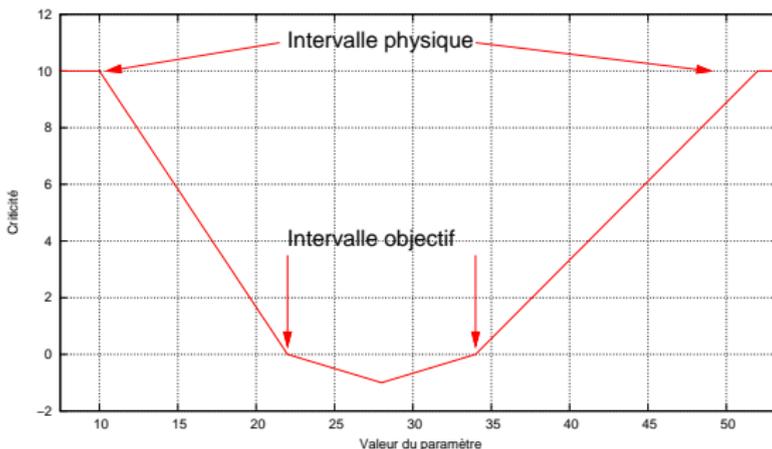
- Sens indirect : « backwardMessage » 

- Sens de la modification
- Criticité de la demande
- Origine de la demande

■ Objectifs individuels

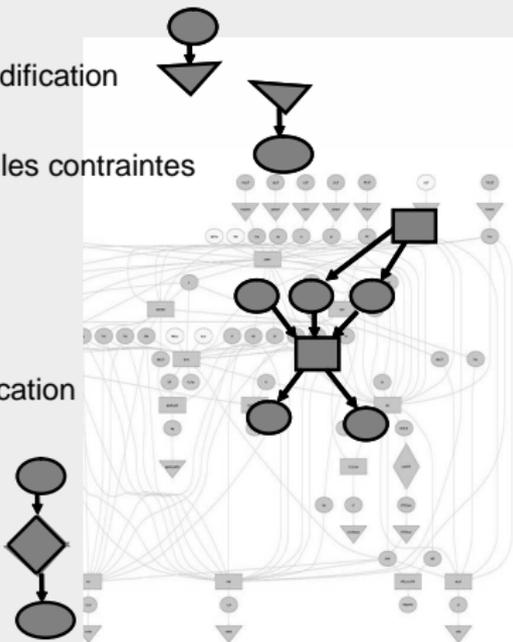
MASCODE : Modèles de connaissances de l'agent

- Les limites physiques d'un paramètre.
- Les valeurs limites de l'objectif.
- Une valeur de criticité calculée



MASCODE : Rôles des agents

- **Agent objectif de sortie**
 - Créer et transmettre des requêtes de modification
- **Agent objectif d'entrée**
 - Adapte la valeur de l'entrée pour réguler les contraintes
- **Agent disciplinaire**
 - Pour chaque entrée
 - Adapter la valeur de l'entrée
 - Ou
 - Transférer une requête de modification
- **Agent cycle**
 - Créer des requêtes de modification
 - Adapter la valeur d'entrée



MASCODE : Raisonnement coopératif

■ Caractéristiques principales :

- Obtenir un consensus sur les valeurs de la conception
- Il agit pour améliorer la situation de son voisin le plus critique
- \Rightarrow Son objectif : diminuer la criticité la plus élevée

■ Processus de traitement des modifications :

- 1 Réception des requêtes provenant des successeurs 
- 2 Comparaison des criticités des demandes de modifications
- 3 Si l'agent est le plus critique alors :
 - Il agit pour lui-même
 - **Sinon** il agit pour les autres
- 4 Envoie ses requêtes 

MASCODE : Besoins de capacités d'adaptation

1 La distribution du raisonnement et des décisions

- Modifications propagées d'un agent à l'autre
- Perception des agents est incomplète = voisinage

2 Structure et les propriétés du problème

- Interdépendances et non-linéarités
- ⇒ Une petite fluctuation sur une entrée a des effets positifs ou négatifs sur les objectifs

⇒ **Le système doit apprendre par interactions et à l'aide des feedbacks des voisins.**

MASCODE : Capacités d'adaptation

1 Pas d'adaptation : $\gamma^{Aq,xi}(t)$

- Objectif : tenir compte du feedback des voisins
- Si **feedback est positif** (demandes successivement dans le même sens)
 - On **augmente le pas** :

$$\gamma^{Aq,xi}(t+1) = \gamma^{Aq,xi}(t) + \psi^{Aq,xi} \quad (1)$$

- **A l'inverse** si les demandes sont opposées
 - On **diminue le pas** :

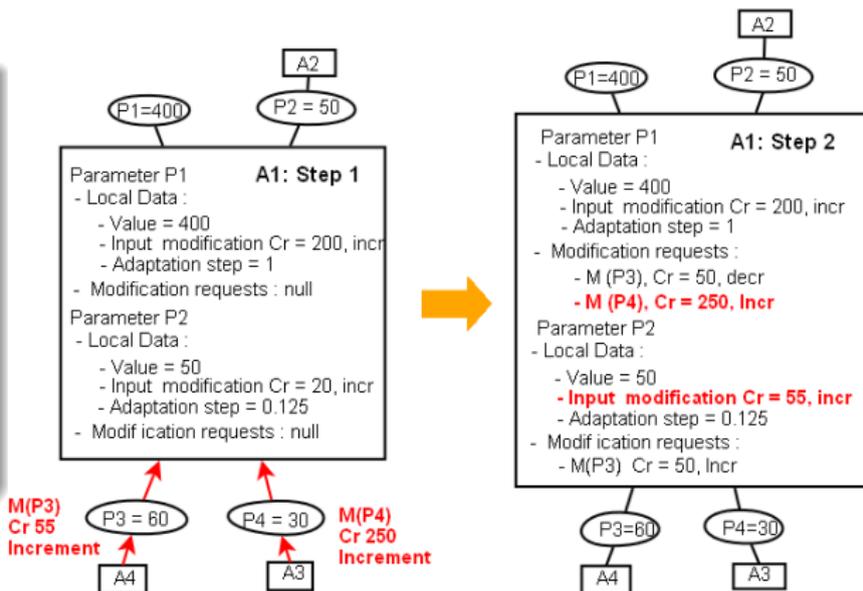
$$\gamma^{Aq,xi}(t+1) = \frac{\gamma^{Aq,xi}(t)}{K} \quad (2)$$

2 Mesure de la confiance

- Objectif : ignorer les demandes de modification provenant d'agents incertains

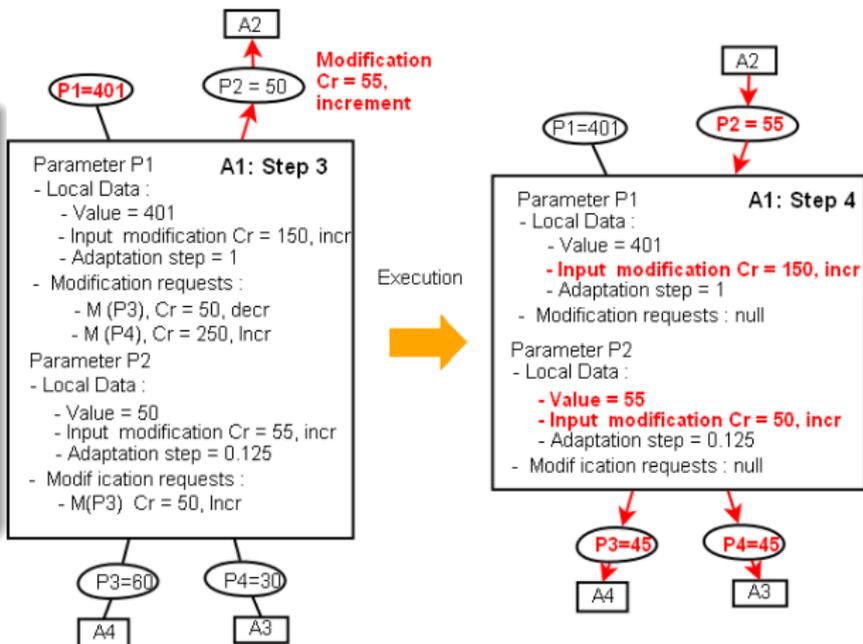
Déroulement d'un scénario (Agent Disciplinaire)

- Réception de messages de modification et sélection des requêtes
- Envoi des messages de modification puis réception des exécutions
- Réception des nouvelles modifications et adaptation du pas de modification
- ...



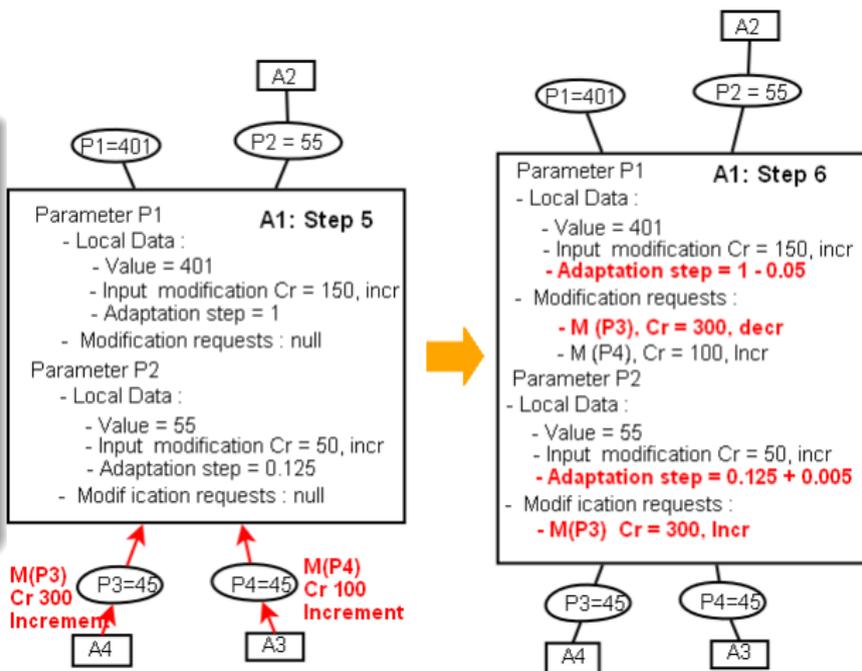
Déroulement d'un scénario (Agent Disciplinaire)

- Réception de messages de modification et sélection des requêtes
- Envoi des messages de modification puis réception des exécutions
- Réception des nouvelles modifications et adaptation du pas de modification
- ...



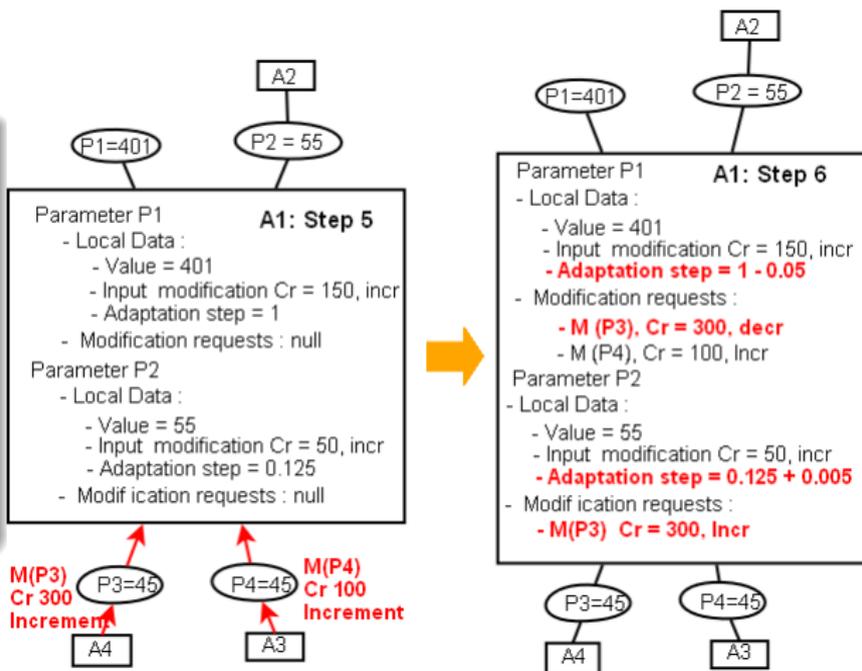
Déroulement d'un scénario (Agent Disciplinaire)

- Réception de messages de modification et sélection des requêtes
- Envoi des messages de modification puis réception des exécutions
- Réception des nouvelles modifications et adaptation du pas de modification
- ...



Déroulement d'un scénario (Agent Disciplinaire)

- Réception de messages de modification et sélection des requêtes
- Envoi des messages de modification puis réception des exécutions
- Réception des nouvelles modifications et adaptation du pas de modification
- ...

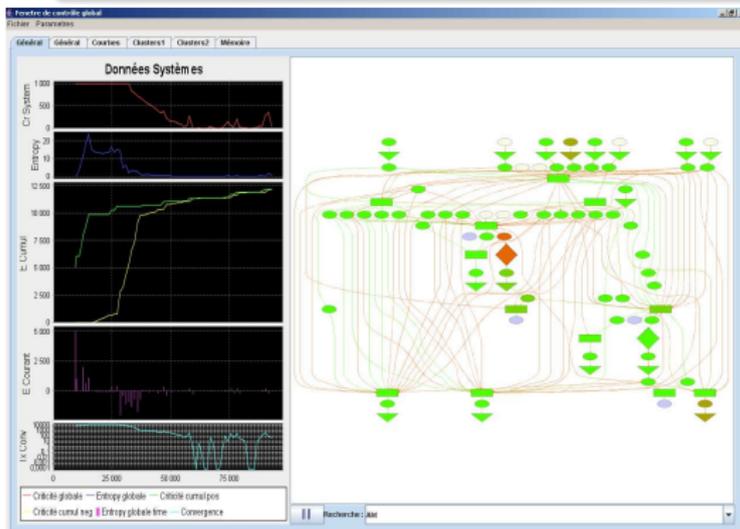


Outline

- 1 Introduction
 - Contexte de la conception avant-projet avion
 - Objectifs de MASCODE
- 2 MASCODE
 - Régulation des contraintes
 - Comprendre le comportement global
 - Optimisation
- 3 Résultats
 - Avantages propres à MASCODE
 - Comparaison sur un cas simple
- 4 Travaux futurs

Analyser le comportement émergent

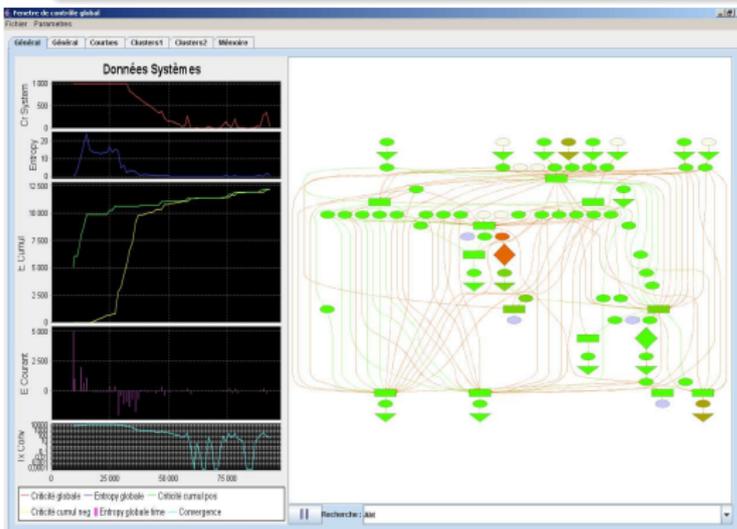
- **Besoin d'une représentation structurelle (réseau)**
- **Besoin d'observer le comportement dynamique (indicateurs)**



- Une vue du réseau d'agents (Agent Observateur)
- Des codes couleurs représentant les agents critiques

Analyser le comportement émergent

- Besoin d'une représentation structurale (réseau)
- Besoin d'observer le comportement dynamique (indicateurs)

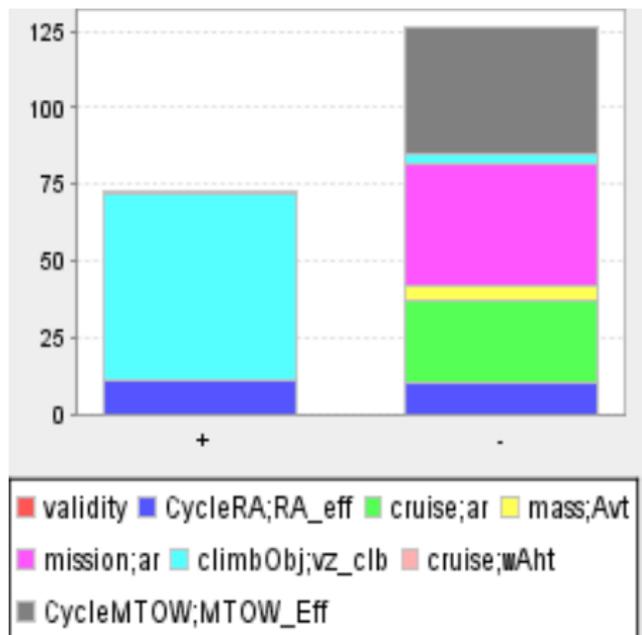


- Criticité globale
 - Criticité de l'agent le plus critique
- Mesure du désordre
 - Agrégation de la dynamique de chaque agent
- Agrégation des criticités absorbées/générées

Identifier les conflits et interdépendances

L'agent architecte mémorise des informations

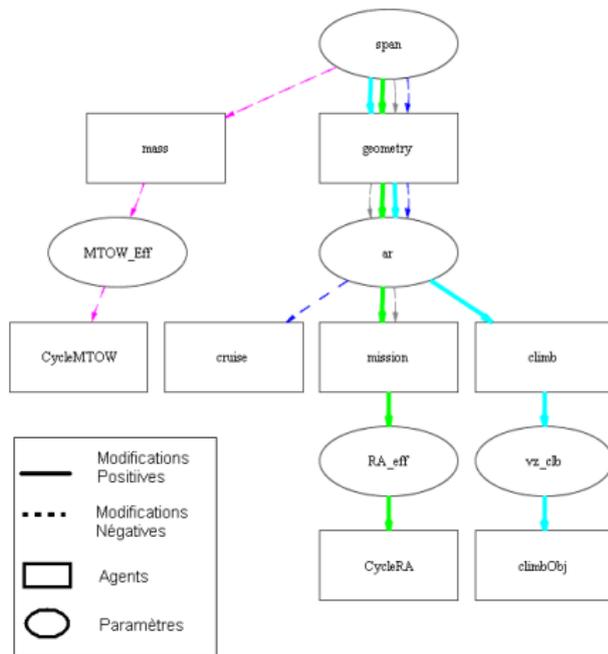
- Les interdépendances d'une entrée par rapport aux sorties (mémoire à long terme)
- La construction des arbres de conflits (mémoire à court terme)
- L'impact d'une modification sur les autres objectifs



Identifier les conflits et interdépendances

L'agent architecte mémorise des informations

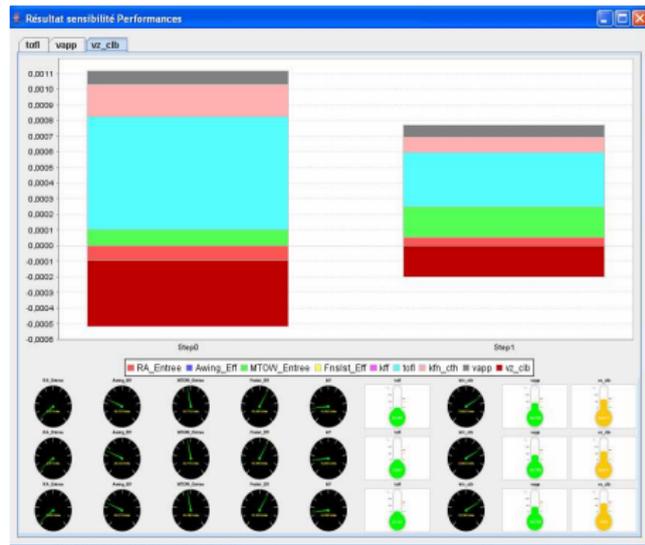
- Les interdépendances d'une entrée par rapport aux sorties (mémoire à long terme)
- La construction des arbres de conflits (mémoire à court terme)
- L'impact d'une modification sur les autres objectifs



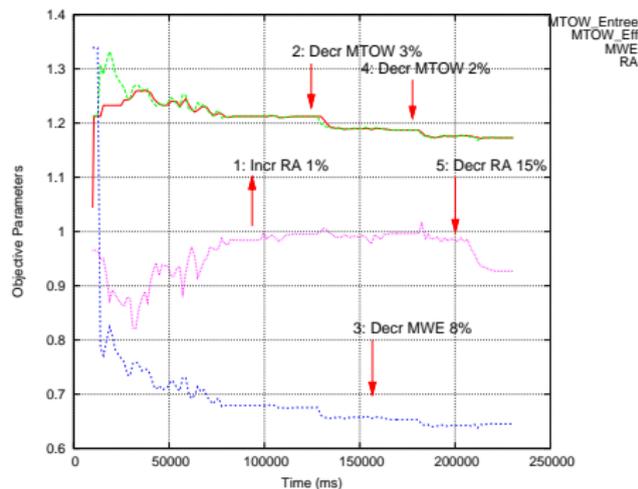
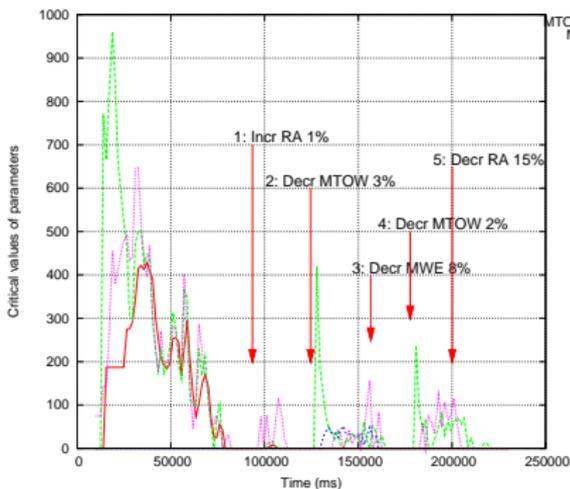
Identifier les conflits et interdépendances

L'agent architecte mémorise des informations

- Les interdépendances d'une entrée par rapport aux sorties (mémoire à long terme)
- La construction des arbres de conflits (mémoire à court terme)
- L'impact d'une modification sur les autres objectifs

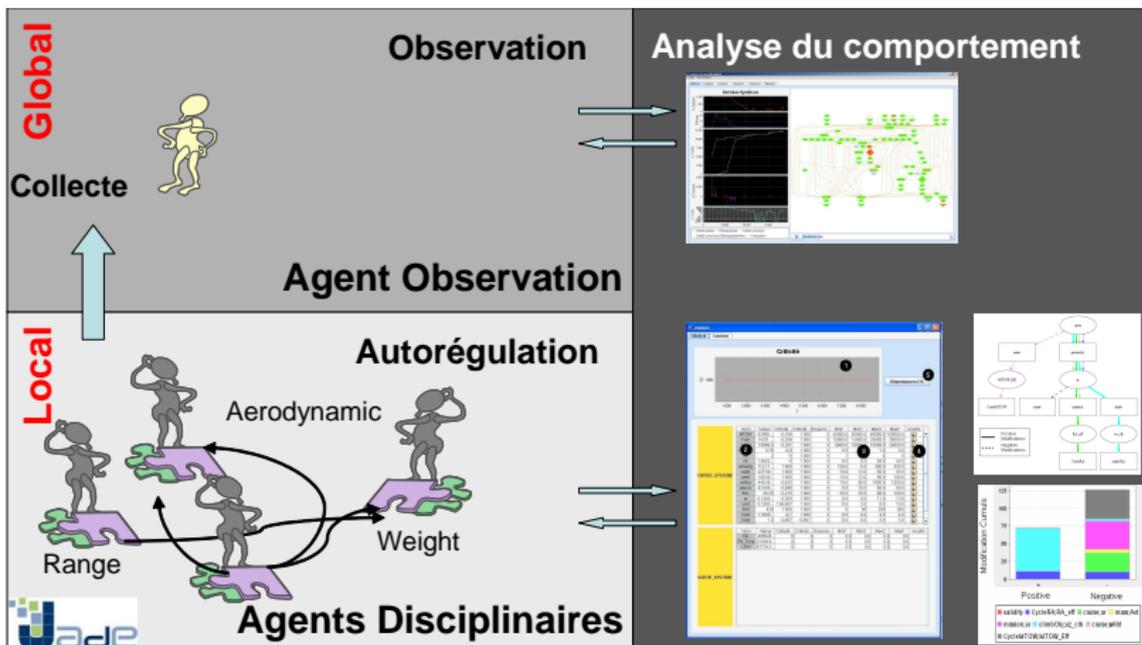


MASCODE : La co-construction des solutions



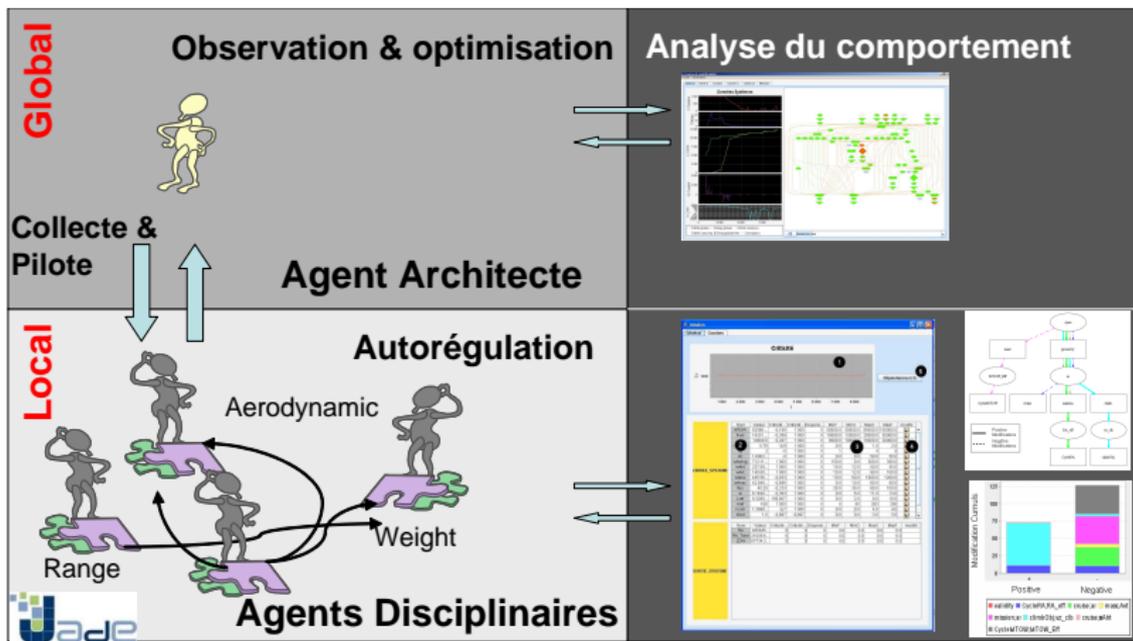
- Criticité = 0 \Rightarrow Toutes les contraintes sont satisfaites
- Conception avion \Rightarrow un processus itératif
- Concepteur affine son problème durant la résolution (co-construction)

Synthèse MASCODE - régulation



- Analyse et observation du système
- Objectif : définir un mode d'action qui trouve des solutions optimales

Synthèse MASCODE - régulation



- Analyse et observation du système
- Objectif : définir un mode d'action qui trouve des solutions optimales

Outline

- 1 Introduction
 - Contexte de la conception avant-projet avion
 - Objectifs de MASCODE
- 2 MASCODE
 - Régulation des contraintes
 - Comprendre le comportement global
 - Optimisation
- 3 Résultats
 - Avantages propres à MASCODE
 - Comparaison sur un cas simple
- 4 Travaux futurs

L'optimisation

- *Pour optimiser une solution, on cherche à améliorer au moins l'un des objectifs*
- ⇒ Nécessite un déplacement dans l'espace des solutions

Dans MASCODE, l'optimisation se fait par Agent Architecte :

- 1 Observe l'état des agents
- 2 Modifie certaines contraintes
- 3 Laisse les agents trouver un nouvel équilibre
- 4 Mémoire les meilleures solutions

L'Agent Architecte ne fait qu'utiliser la capacité d'adaptation de système, il ne participe pas à la résolution

L'optimisation

- *Pour optimiser une solution, on cherche à améliorer au moins l'un des objectifs*
- ⇒ Nécessite un déplacement dans l'espace des solutions

Dans MASCODE, l'optimisation se fait par Agent Architecte :

- 1 Observe l'état des agents
- 2 Modifie certaines contraintes
- 3 Laisse les agents trouver un nouvel équilibre
- 4 Mémorise les meilleures solutions

L'Agent Architecte ne fait qu'utiliser la capacité d'adaptation de système, il ne participe pas à la résolution

L'optimisation

- *Pour optimiser une solution, on cherche à améliorer au moins l'un des objectifs*
- ⇒ Nécessite un déplacement dans l'espace des solutions

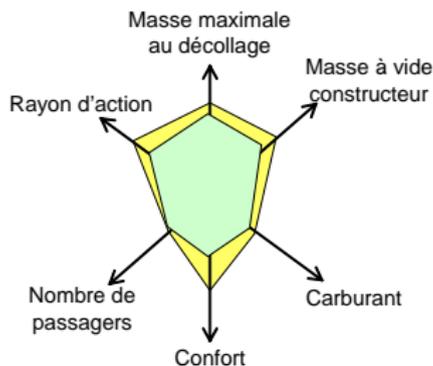
Dans MASCODE, l'optimisation se fait par Agent Architecte :

- 1 Observe l'état des agents
- 2 Modifie certaines contraintes
- 3 Laisse les agents trouver un nouvel équilibre
- 4 Mémorise les meilleures solutions

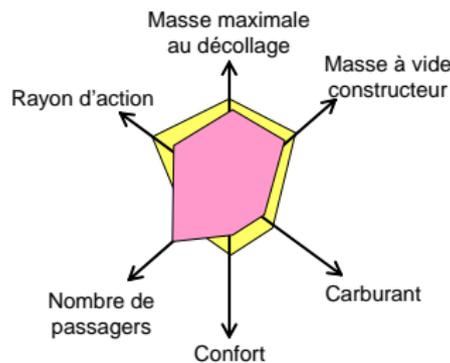
L'Agent Architecte ne fait qu'utiliser la capacité d'adaptation de système, il ne participe pas à la résolution

Stratégie pour obtenir une solution de Pareto

- Multi-objectif \Rightarrow de **multiples solutions**,
- \Rightarrow On cherche des solutions de **Pareto**, pour lesquelles **l'amélioration d'un objectif** ne peut se faire sans la **dégradation d'au moins un des autres objectifs**.



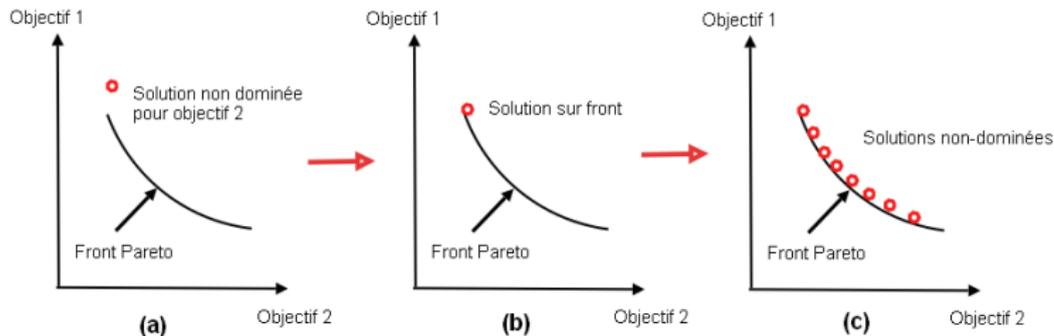
La configuration jaune domine la configuration verte



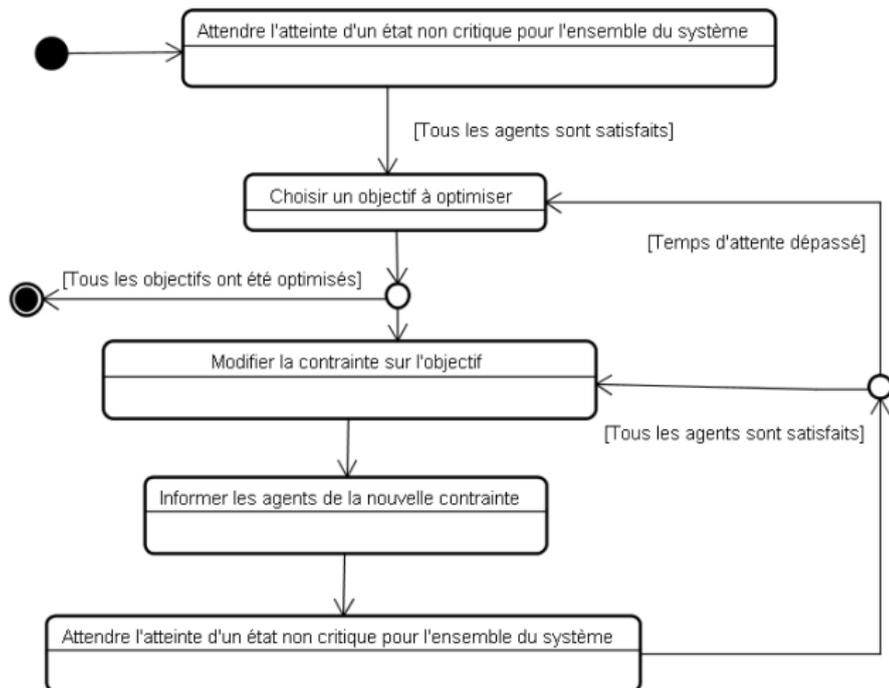
Aucune configuration ne domine l'autre

Stratégie pour obtenir une solution de Pareto

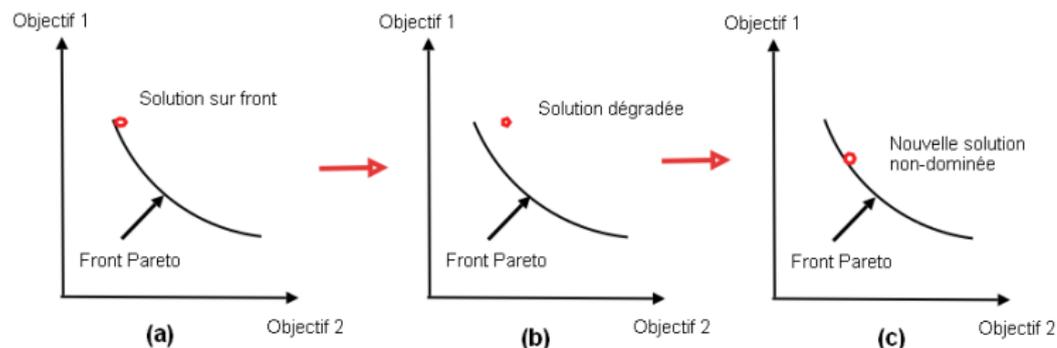
- Multi-objectif \Rightarrow de **multiples solutions**,
- \Rightarrow On cherche des solutions de **Pareto**, pour lesquelles l'**amélioration d'un objectif** ne peut se faire sans la **dégradation d'au moins un des autres objectifs**.



Stratégie pour obtenir une solution de Pareto

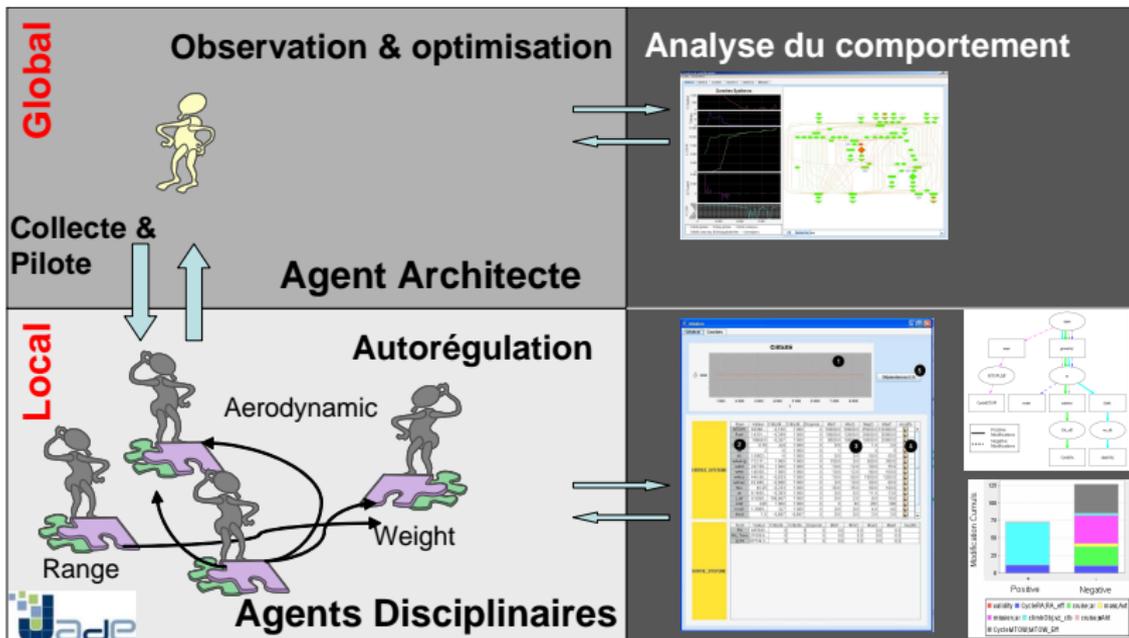


Stratégie pour le parcours de solutions de Pareto



- 1 On part d'une solution de Pareto (a)
- 2 On dégrade au moins un des objectifs (b)
- 3 On optimise d'autres objectifs que celui ou ceux dégradé(s) (c).

Synthèse MASCODE - Optimisation



Plan

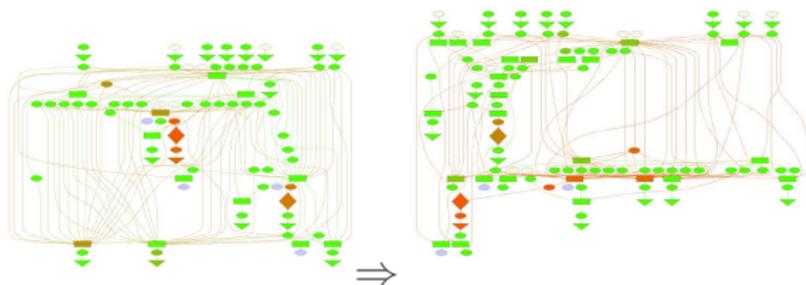
- 1 Introduction
 - Contexte de la conception avant-projet avion
 - Objectifs de MASCODE
- 2 MASCODE
 - Régulation des contraintes
 - Comprendre le comportement global
 - Optimisation
- 3 Résultats
 - Avantages propres à MASCODE
 - Comparaison sur un cas simple
- 4 Travaux futurs

Augmentation du nombre d'agents

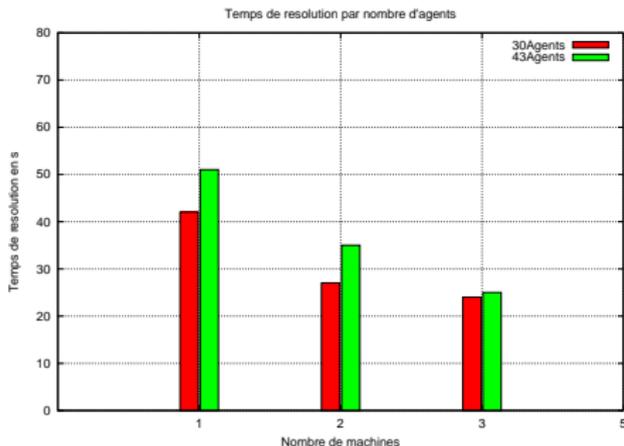
En satisfaction de contraintes

- On choisit et remplace une fonction à décomposer, e.g. la *masse*
- La nature du problème est identique

	Test 1	Test 2
Agent	30	43
nb var. effectives	62	74
nb var. en entrée	15	17
nb var. en sortie	9	8
nb de bouclages	2	2
nb de critères contraints	49	59
nb de critères de sorties	5	5
nb de constantes	7	9



Résultats : augmentation du nombre d'agents



Résultats

- L'augmentation des agents implique une augmentation du temps de convergence
- L'augmentation est négligeable dès que l'on distribue les agents

Conclusion

L'augmentation seule du nombre d'agent ne semble pas un facteur de complexité

Modification du nombre de degrés de liberté

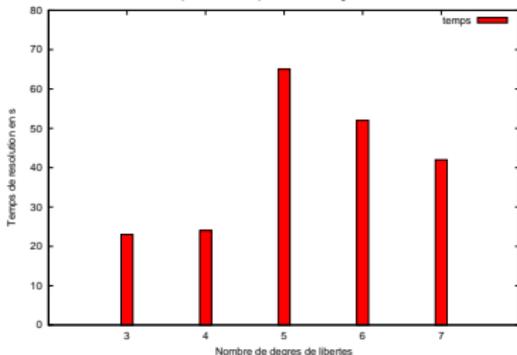
Objectif : Étudier l'influence du nb de degrés de liberté

- Pour un problème donné, on fait varier le nombre de degrés de liberté
- Lorsque l'on retire un critère, on lui donne une valeur solution (trouvée à l'étape précédente)

Nombre de degrés de liberté	Élément fixé
7	N/A
6	span
5	span tuc
4	fuel span tuc
3	phi fuel span tuc

Modification du nombre de degrés de liberté

Temps de resolution par nombre de degres de liberte



Résultats

- Le temps de convergence n'augmente pas forcément avec le nombre de degrés de liberté
- On suppose qu'il dépend du nombre de conflits ?

Conclusion

⇒ Mascode n'est pas directement dépendant du nombre de degrés de liberté

Quelques conclusions sur le comportement

- Lorsque le nombre d'entrées est important :
 - il existe plus de moyens pour résoudre les conflits
 - il existe aussi potentiellement plus d'actions inutiles
- La complexité de la résolution dépend :
 - des conflits rencontrés par certains paramètres de conception
 - des remises en cause éventuelles sur des choix précédemment opérés

Quelques conclusions sur le comportement

- Lorsque le nombre d'entrées est important :
 - il existe plus de moyens pour résoudre les conflits
 - il existe aussi potentiellement plus d'actions inutiles
- **La complexité de la résolution** dépend :
 - **des conflits rencontrés** par certains paramètres de conception
 - **des remises en cause éventuelles** sur des choix précédemment opérés

Comparaison des résultats avec un Algo. Génétique

Objectif : Montrer que l'on peut aller chercher des optimum

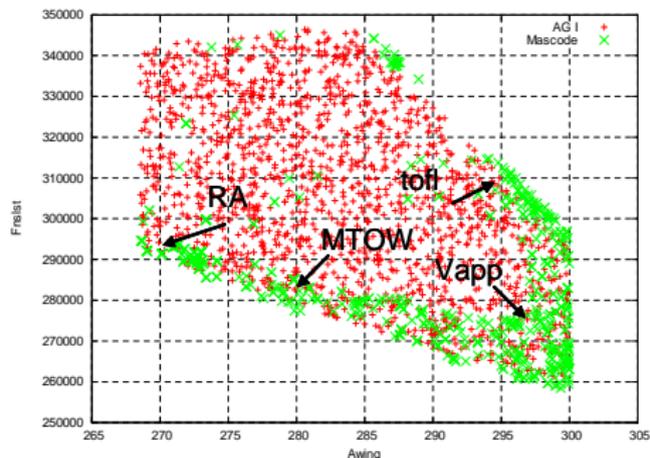
Caractéristiques du problème :

- 2 degrés de liberté : Awing Fnlst
- 5 performances avec contrainte
- 4 performances avec un objectif (Vapp, tofl, MTOW, RA)

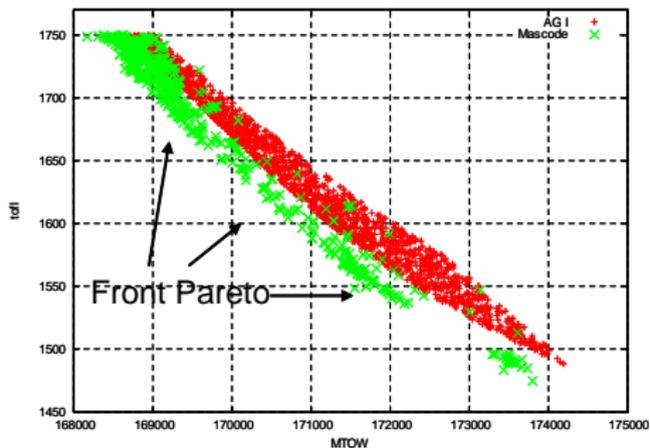
Solution MASCODE en vert

MASCODE a tendance à attirer les solutions vers les frontières du fait de notre stratégie de parcours

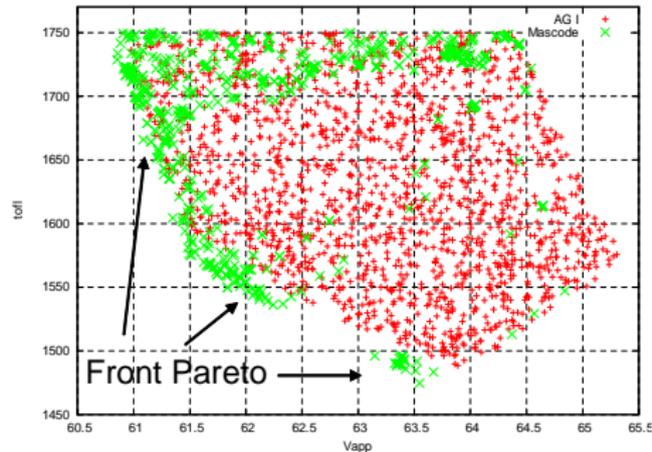
La structure du problème est idéale pour l'AG



Deux fronts de Pareto



Objectifs *MTOW-toff*



Objectifs *Vapp-toff*

Les solutions sont bonnes mais légèrement différentes (meilleures)

- les boucles de rétroaction autorisent 0,2% de différence

Les solutions sont parfois incomplètes

- la stratégie de parcours est basique

Avantages Principaux

- La distribution du raisonnement + utilisation de la structure du problème
- \Rightarrow des caractéristiques intéressantes

- **Chaque discipline est modélisée indépendamment**
- Le **système est adaptatif et dynamique**
 - **Seules les parties concernées** sont impactées par les **modifications**
- L'approche est **peu sensible au nombre de degrés de liberté**
 - La difficulté \Leftrightarrow nombre de conflits rencontrés
 - Travailler sur des problèmes sous-contraints
- **Peu sensible à l'augmentation du nombre d'agent**

Avantages Principaux

Concepteur

- **Interagir avec le système**
 - Changer les contraintes \Rightarrow un nouvel équilibre dynamique
- **Problème sur-contraint \Leftrightarrow répartition des criticités**
- **Visibilité sur les relations entre disciplines**
 - Le système est considéré comme "une boîte blanche"

	Consistance de la résolution	Sensibilité au nombre de degrés de liberté	Interaction utilisateur	Stratégie
MASCODE	A la fin lorsque les criticités ont été absorbées	Independent (a priori)	L'utilisateur peut interagir	Propriétés du système, Apprentissage des interdépendances.
AG avec analyseur (MDF)	A chaque évaluation	Fortement dépendant (analyse combinatoire)	Nouvelle résolution	Apprentissage à partir d'expériences aléatoires, puis apprentissage.

Démonstration MASCODE

Plan

1 Introduction

- Contexte de la conception avant-projet avion
- Objectifs de MASCODE

2 MASCODE

- Régulation des contraintes
- Comprendre le comportement global
- Optimisation

3 Résultats

- Avantages propres à MASCODE
- Comparaison sur un cas simple

4 Travaux futurs

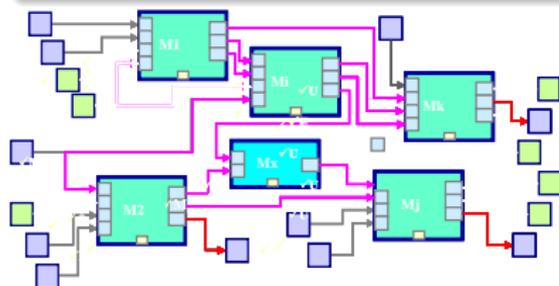
Travaux futurs

- **Ajouter des approches/connaissances mathématiques**
 - outils d'optimisation, gestion d'incertitudes, simplification modèles...
 - étudier la complexité et la convergence du système
- **Intégrer plusieurs niveaux de modélisation** \mapsto **autres aspects métiers**
- **Travailler sur de la modélisation d'architectures systèmes** \mapsto **généraliser**

Travaux futurs

D'un point de vue agent :

- **Permettre la gestion d'organisations dynamiques** (concurrence sur la précision, le temps d'exécution, les hypothèses physiques...)
- **Tester d'autres problèmes** (familles d'avions, approches multi-utilisateurs...)



Auto-régulation