

# Diverses techniques d'optimisation inspirées de l'auto-organisation dans les systèmes biologiques

**Johann Dréo & Patrick Siarry**

Université Paris 12 (LERISS)

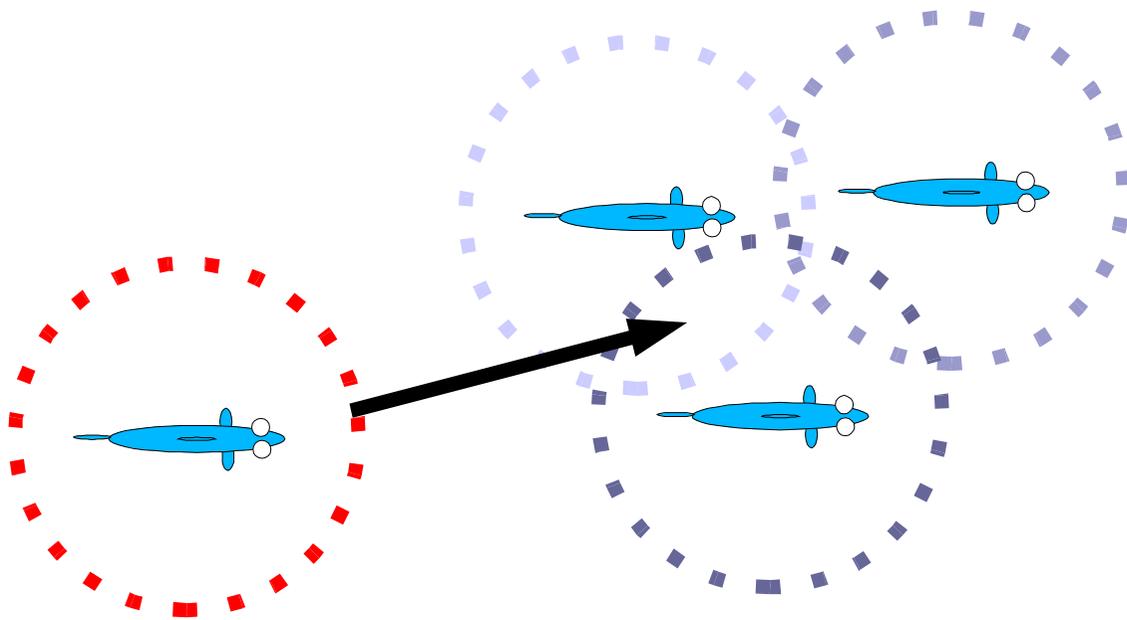
*Séminaire Optimisation par Essaim Particulaire*

*Paris*

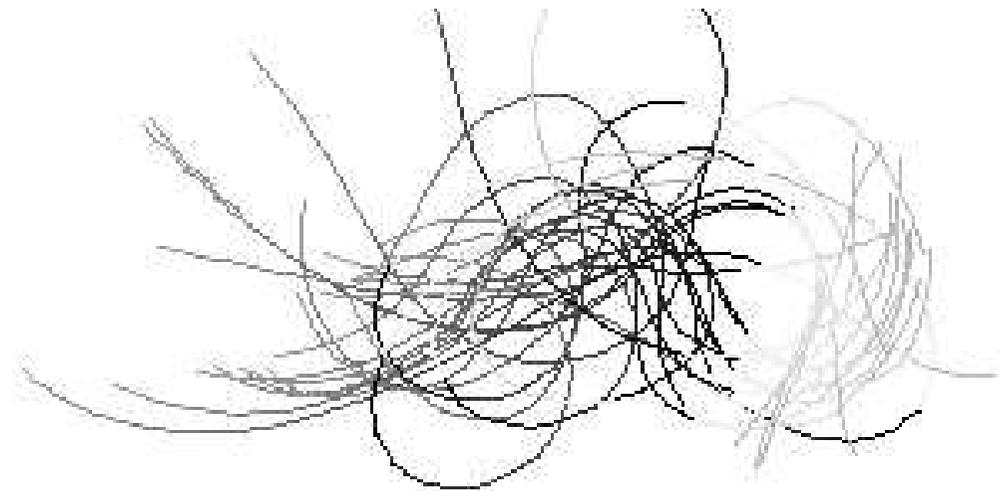
# Plan

- Métaheuristiques :
  - OEP,
  - Algorithmes évolutionnaires,
  - Systèmes immunitaires,
  - Colonies de fourmis
- Théories :
  - Auto-organisation,
  - Programmation à mémoire adaptative,
  - Bases communes.
- HCIAC (Colonie de fourmis + communication) :
  - Hétérarchie dense & communication,
  - Bases, Hybridation,
  - Résultats.
- Conclusion.

# OEP

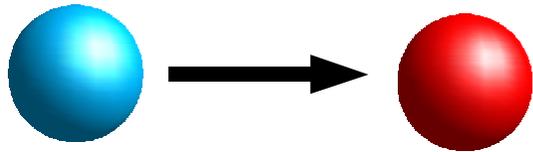


- Optimisation
- Classification
- ...

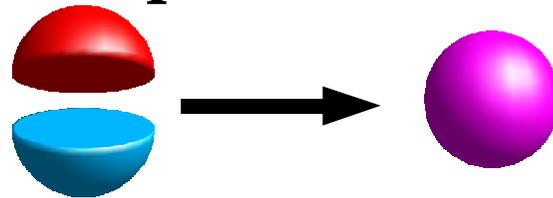


# Algorithmes évolutionnaires

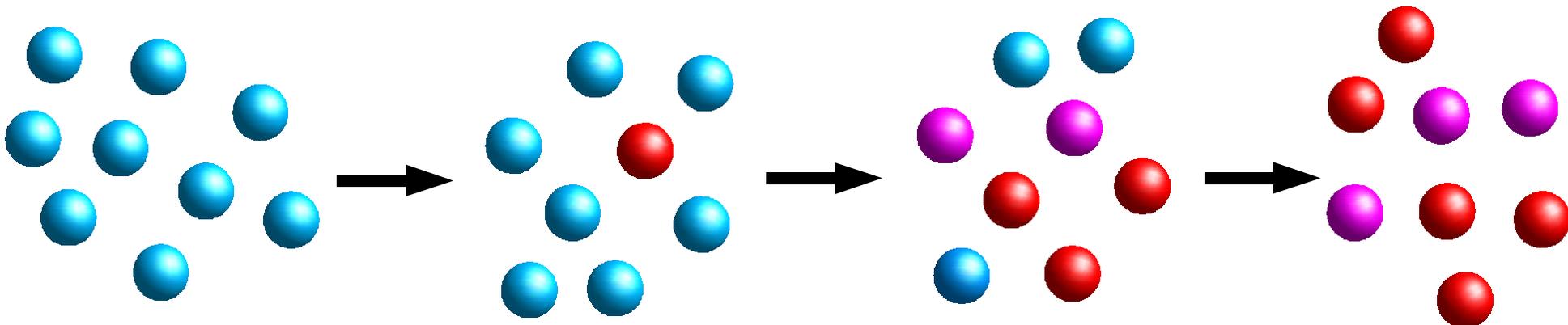
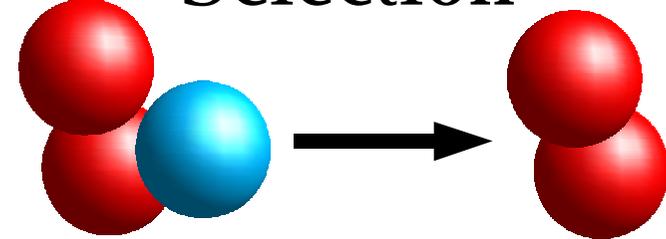
Mutation



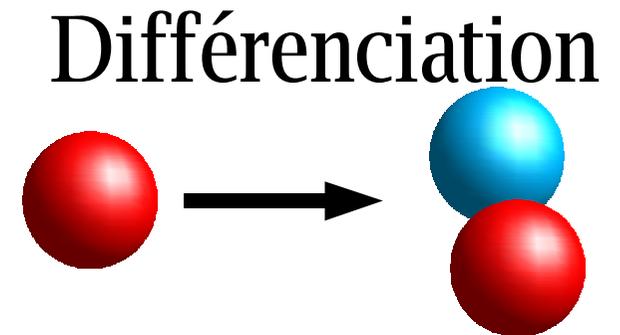
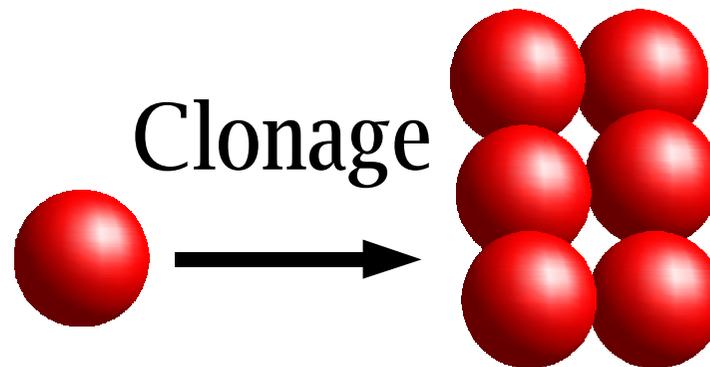
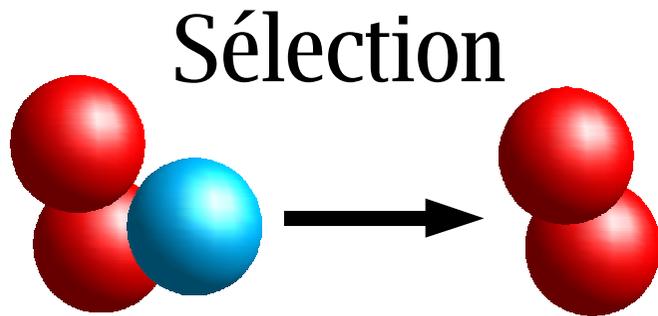
Reproduction



Sélection

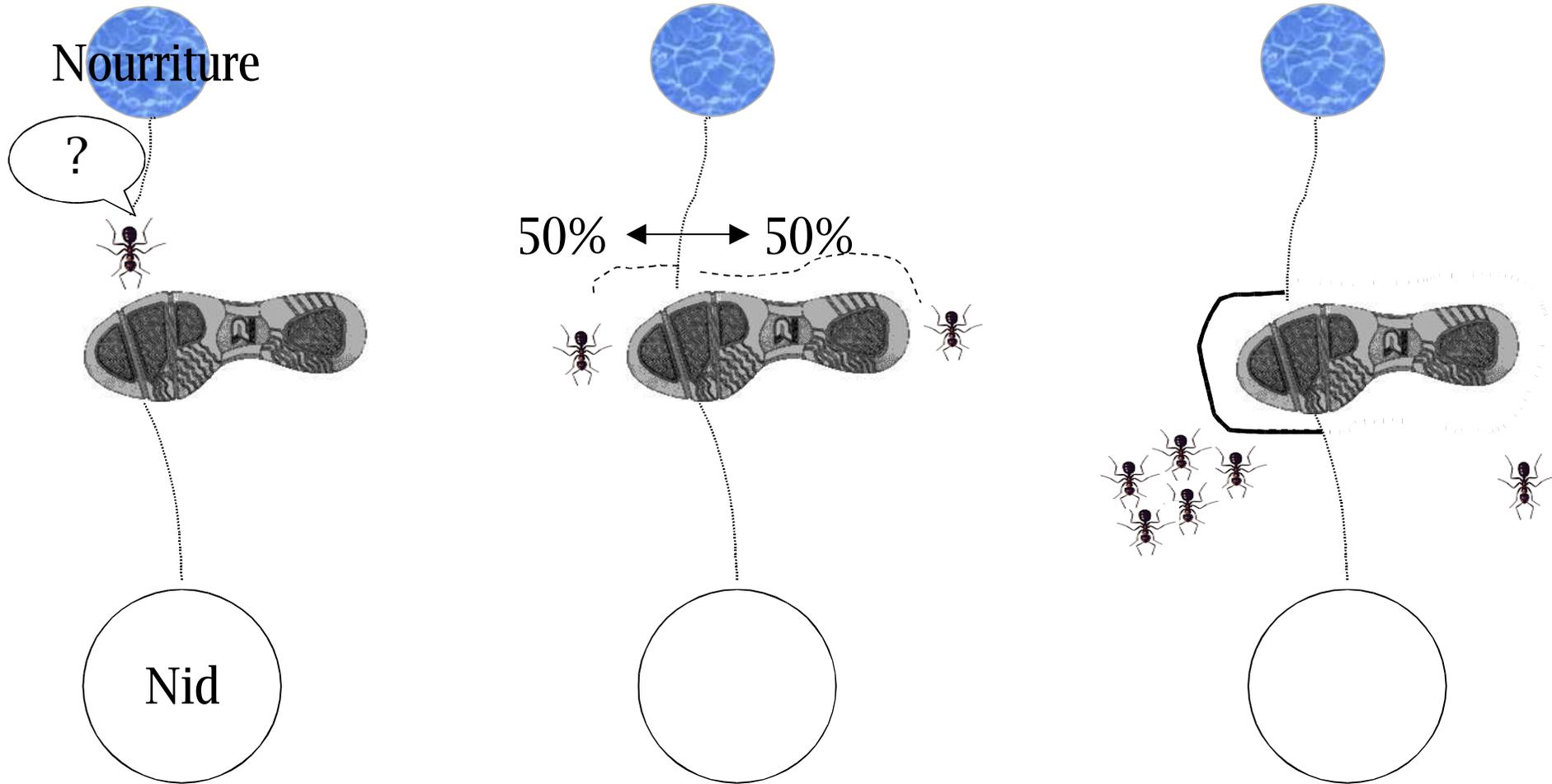


# Systemes immunitaires



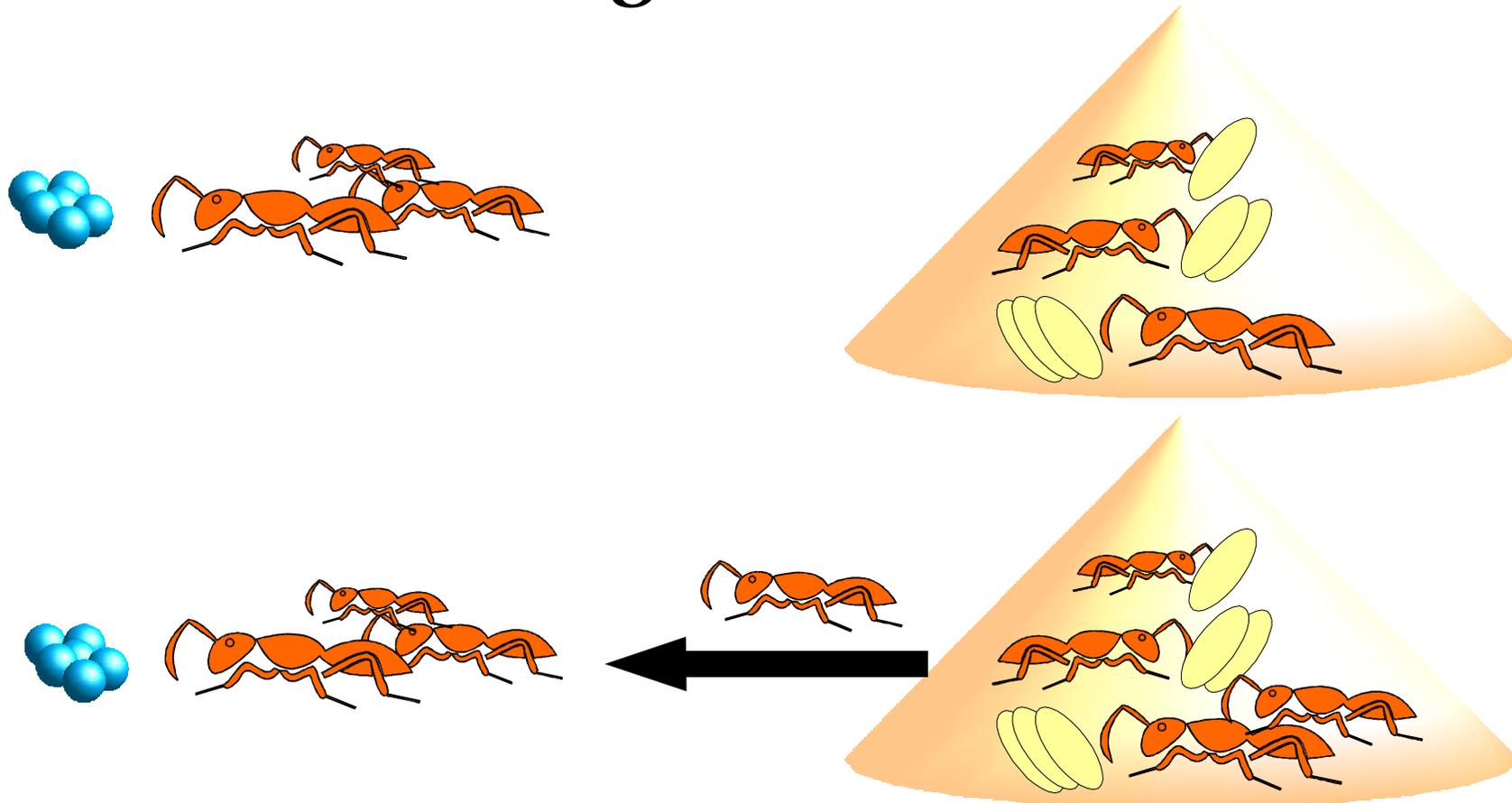
**Artificial Immune Systems and their applications**  
*Dasgupta,*  
Springer Verlag, 1999

# Colonies de fourmis



## Optimisation discrète

# Fourmis et organisation du travail



## Allocation dynamique de tâches

# Auto-organisation

L'auto-organisation est un **processus** dans lequel un **modèle** de niveau **global émerge** uniquement d'un grand nombre d'**interactions** entre les **composants** de bas niveau du système.

De plus, les règles spécifiant les interactions entre composants du système sont suivies en utilisant uniquement des **informations locales**, sans référence au modèle global.

## **Self-Organization in Biological Systems**

*Camazine, Deneubourg, Franks, Sneyd, Theraulaz and Bonabeau*

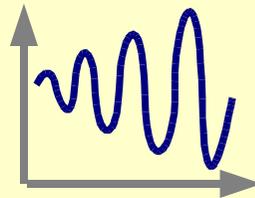
Princeton University Press, 2000

# Auto-organisation

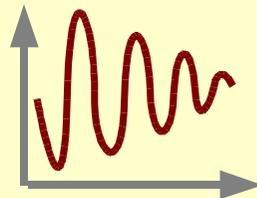
Composants **simples**  $\xrightarrow{\text{Émergence}}$  Système **complexe**  
(interactions)

## Rétroactions :

Positives  
Amplification

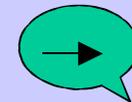


Négatives  
Stabilisation

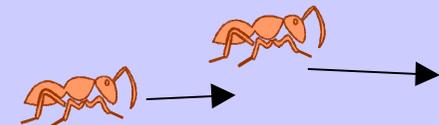


## Flux d'informations :

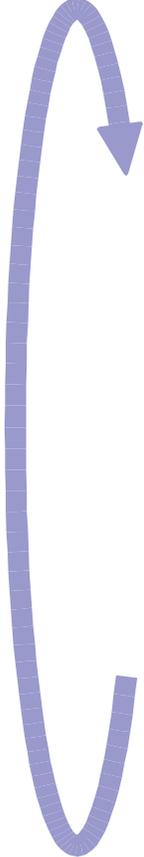
Signaux



Indices



# Programmation à mémoire adaptative



**Information** = Mémoire  
*(Représentation infos)*

**Intensification** = Recherche **locale**  
*(Exploitation infos)*

**Diversification** = Recherche **globale**  
*(Recherche infos)*

**Adaptive Memory Programming: A Unified View of Meta-Heuristics**

*Taillard, Gambardella, Gendreau, Potvin*

European Journal of Operational Research, 1998

# Bases en commun

## BUT

Programmation  
à mémoire adaptative:

*Intensification*  
*Diversification*

## MOYEN

Auto-organisation :

*Rétroactions*  
*Diversification*

Fourmis :

- Regroupement
- Aléatoire / phéromones

OEP :

- Regroupement
- Coopération

Émergence



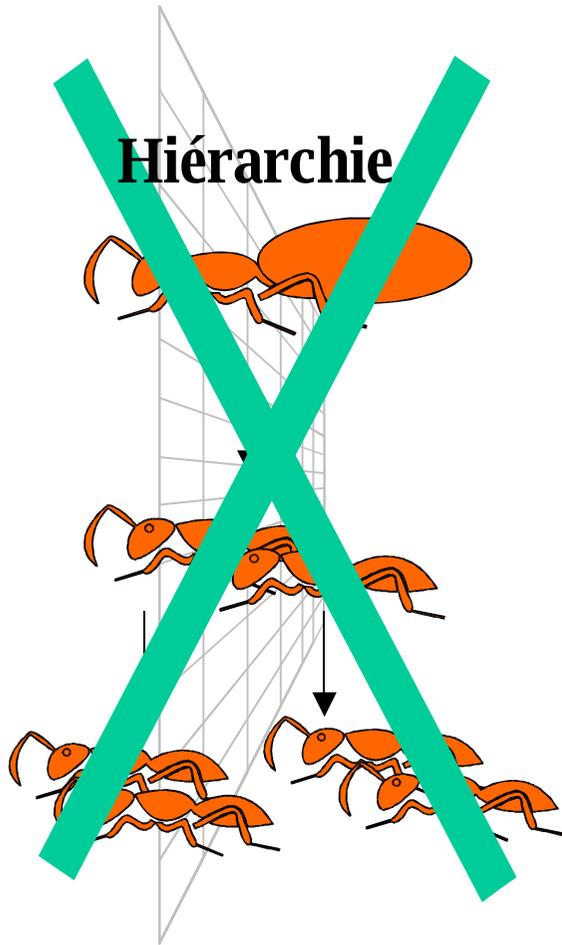
Fourmis :

- Renforcement / saturation
- Pistes

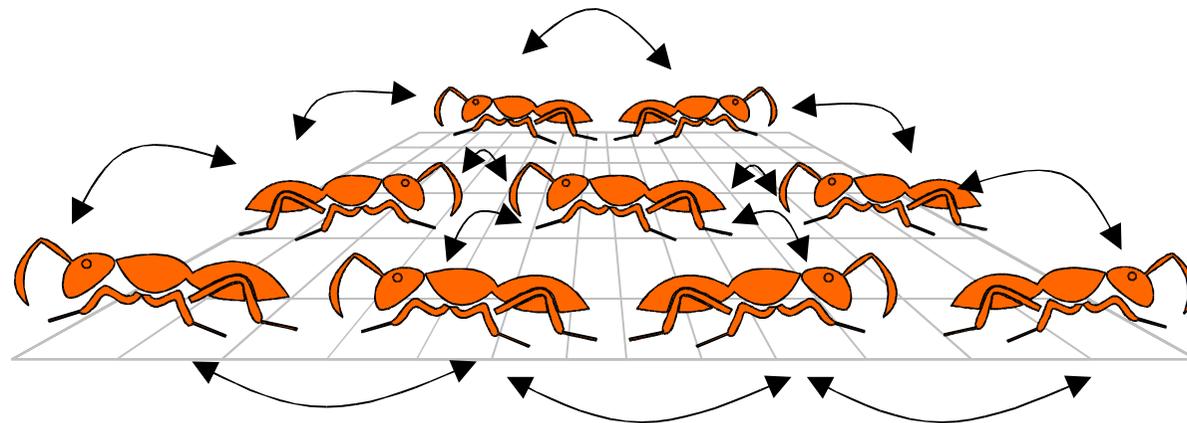
OEP :

- Attirance / inertie
- Valeur  $f_{OBJ}$

# Hétérarchie dense



## Hétérarchie



**Dense Heterarchy and mass communication as the basis of organization in ant colonies.**

*Wilson & Hölldobler,*

*Trends in Ecology and Evolution, 1988*

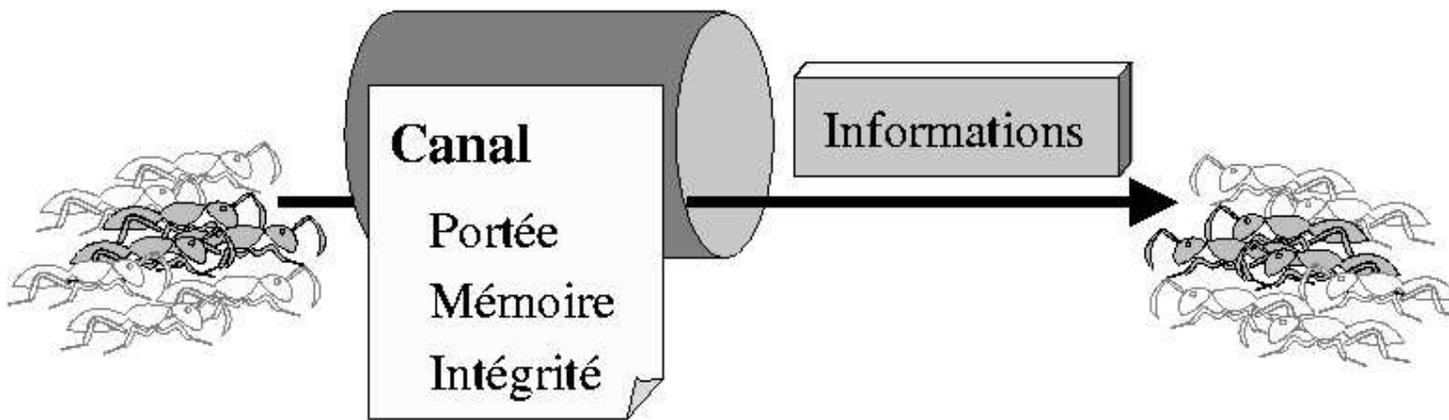
OEP'03 – Jeudi 2 octobre 2003

# Communication

**Informations** {  
Nourriture  
Défense  
...

**Canaux** {  
Pistes  
Trophallaxies  
...

**Propriétés** {  
Indirecte  
Mémoire  
Modifiée  
...

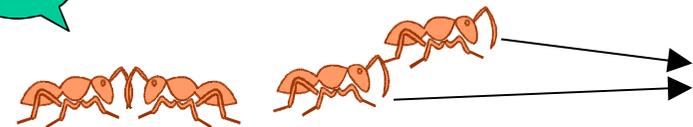


# L'algorithme CIAC

« *Continuous Interacting Ant Colony* »

2 canaux

- Pistes de phéromone
- Recrutement direct



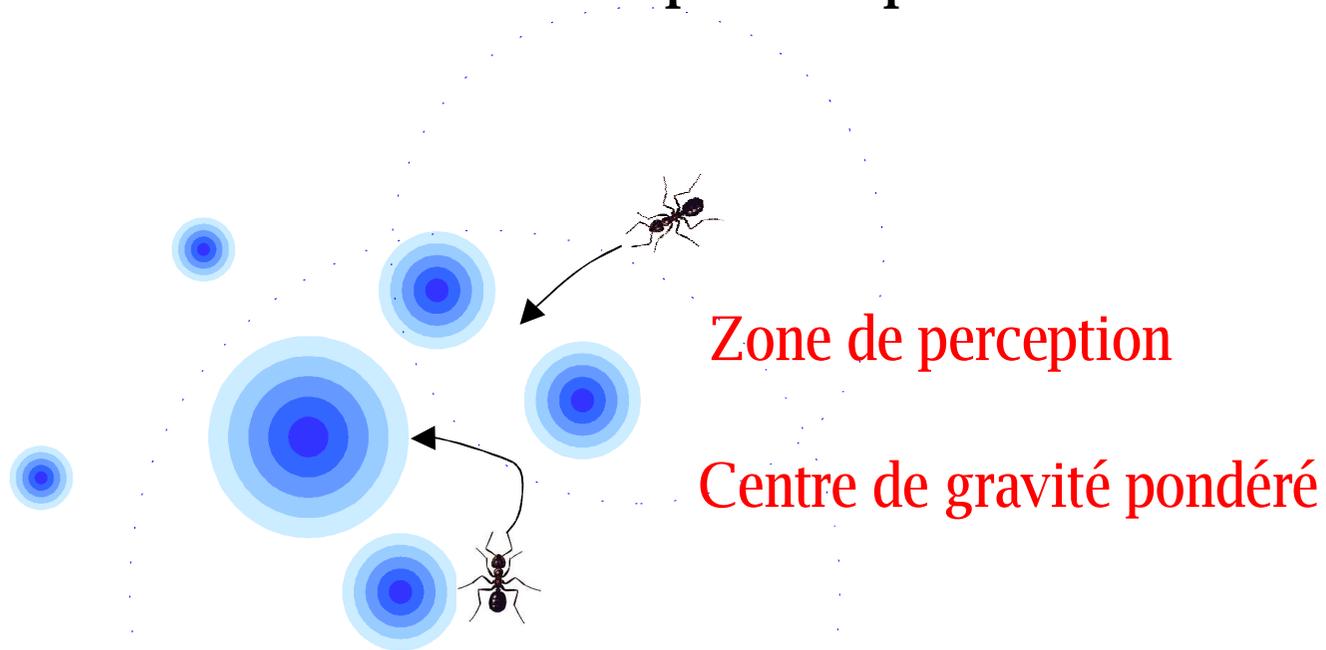
**Continuous Interacting Ant Colony Algorithm**  
**Based on Dense Heterarchy**

*Dréo & Siarry*

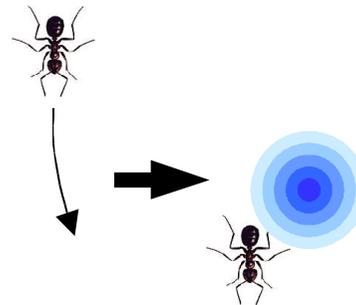
Future Generation Computer Systems, à paraître

# CIAC : base

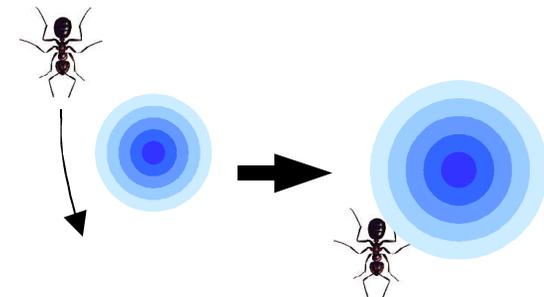
Spots de phéromone



Dépôt



Renforcement

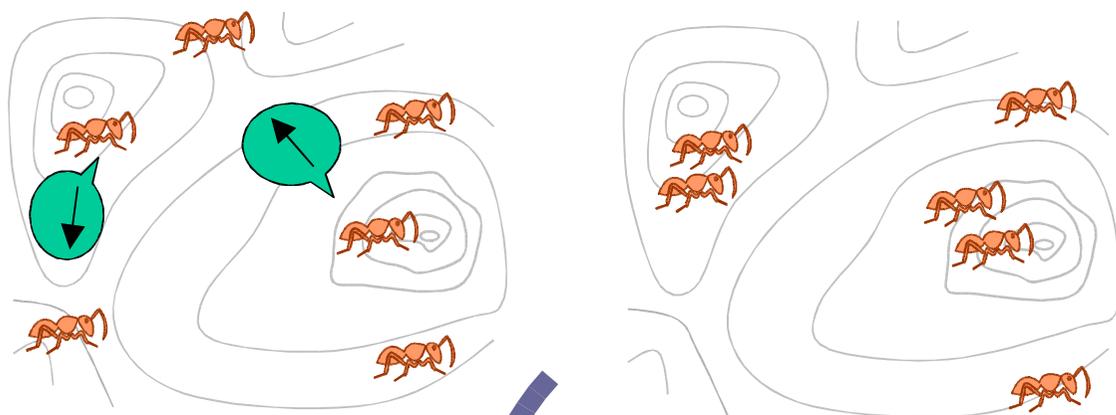


# CIAC : base

Communication directe interindividuelle

Informations :

**Messages** ↔ *Amélioration fonction objectif*  
*Position région d'intérêt*

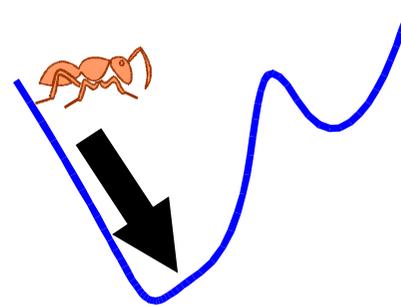


Echange direct d'infos  
Sortie des minimums locaux

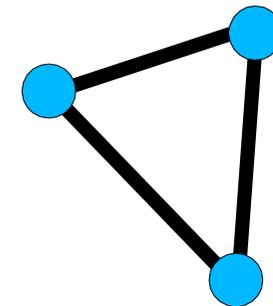
**OEP**

# HCIAC

CIAC + Recherche locale



(Simplex de Nelder-Mead)

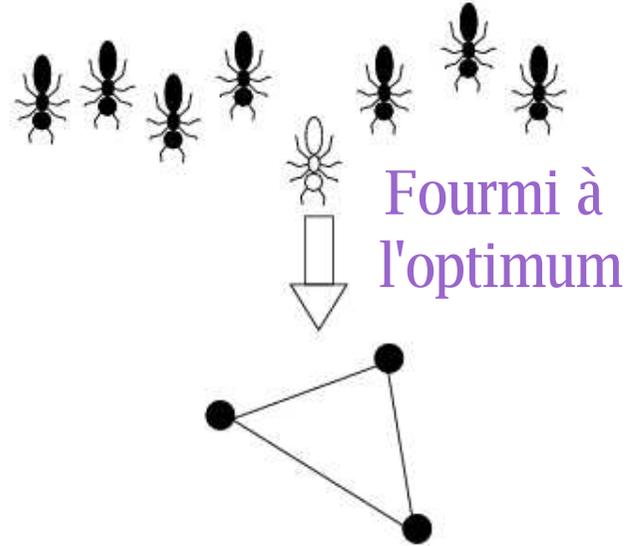


= Algorithme HCIAC

# HCIAC : hybridations simples

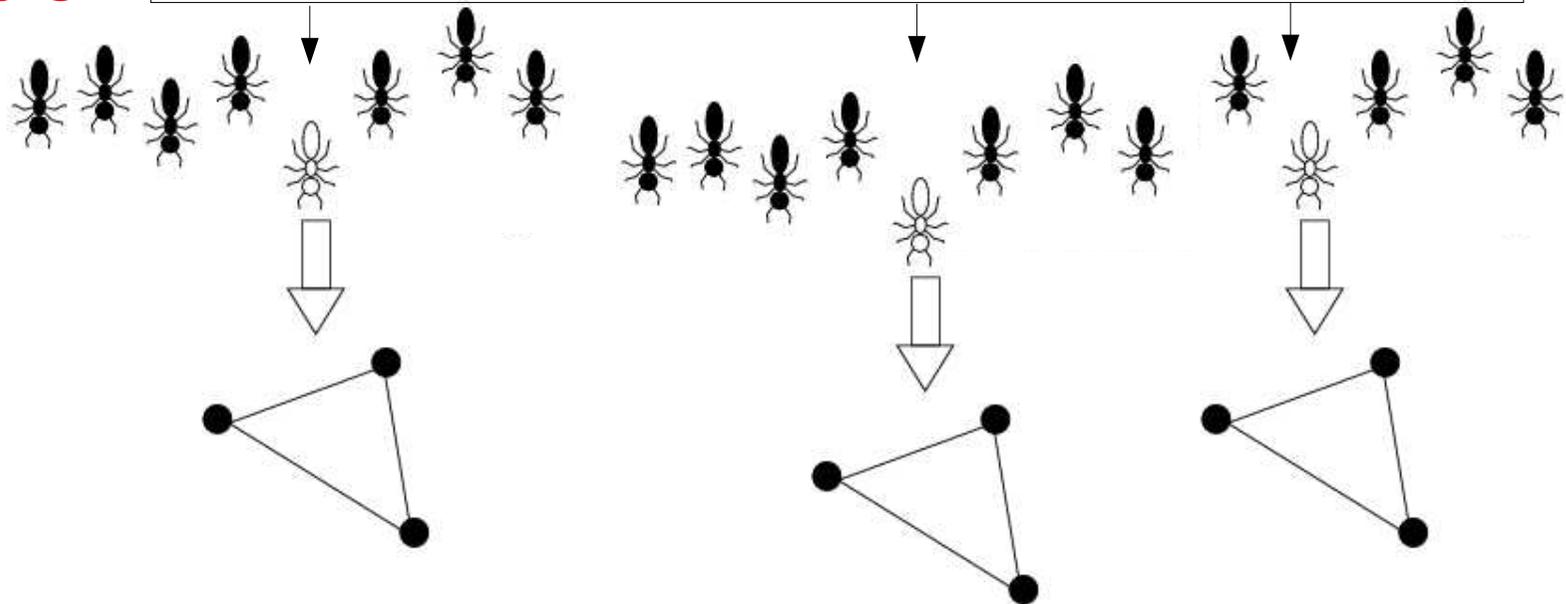
Terminale

Algorithmes...



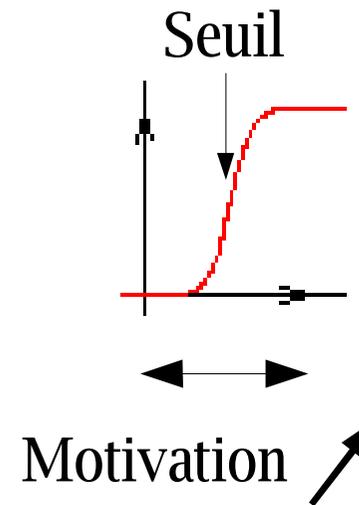
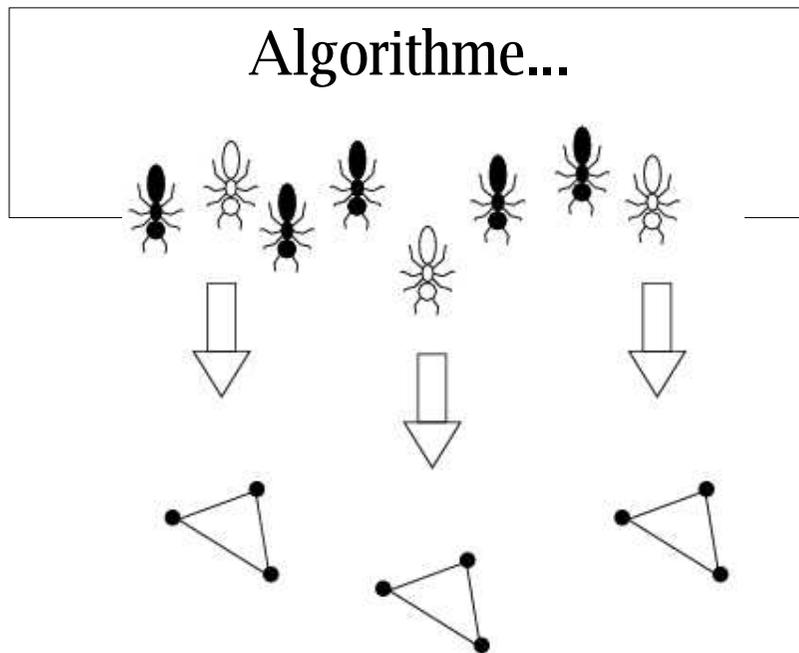
Périodique

Algorithmes...



# HCIAC : hybridation décentralisée

Déclenchement simplex ?



# HCIAC : paramètres

## *Paramétrage*

**Paramètres fournis : distribution normale**

**Paramètres algorithme :**

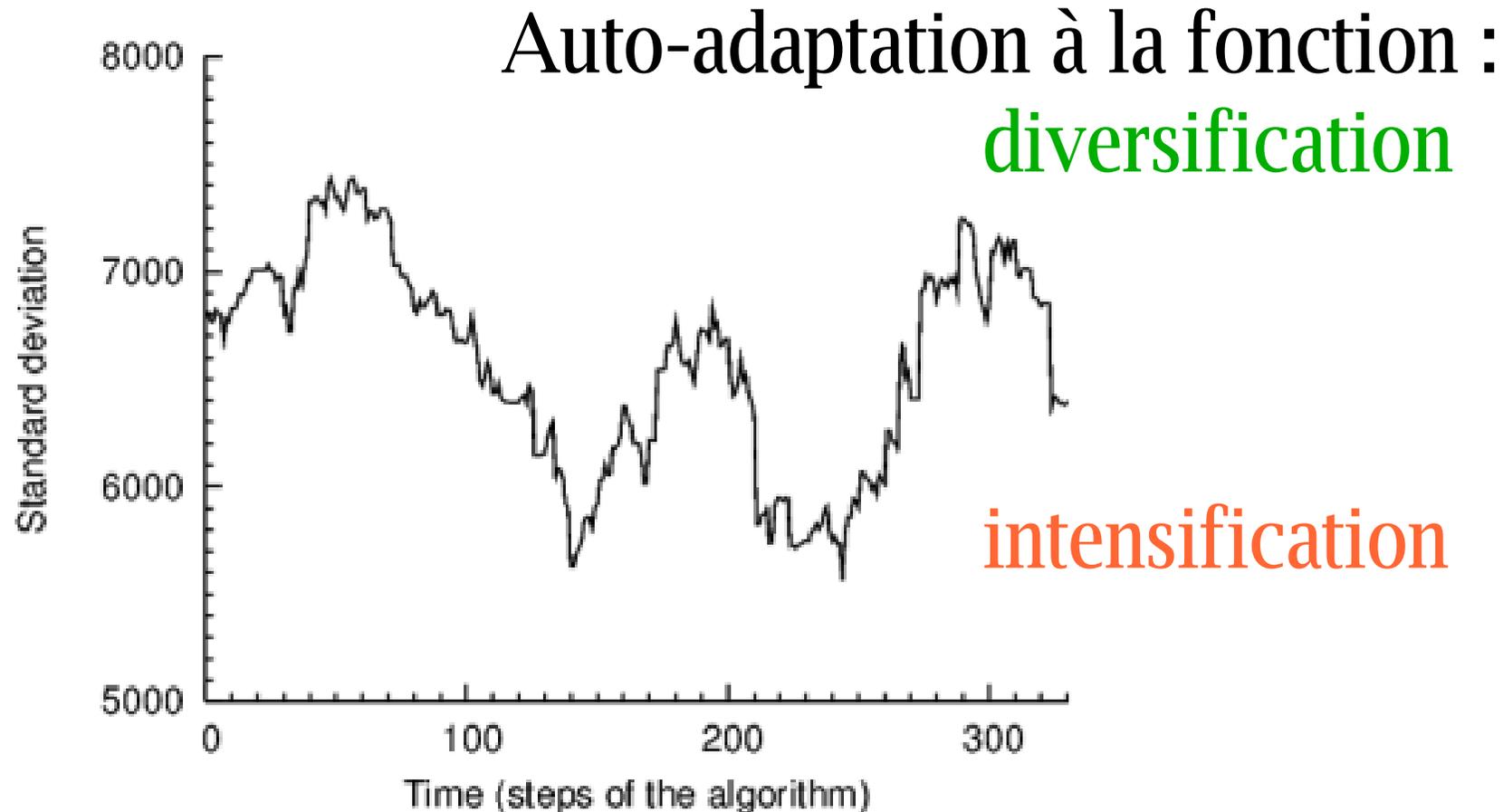
Méta-paramétrage par algo bi-objectif :

1) vitesse

2) précision

**Indice diversification / intensification**

# HCIAC : résultats



# HCIAC : résultats

## Qualités :

- Précision
- Efficacité
- Parallélisme

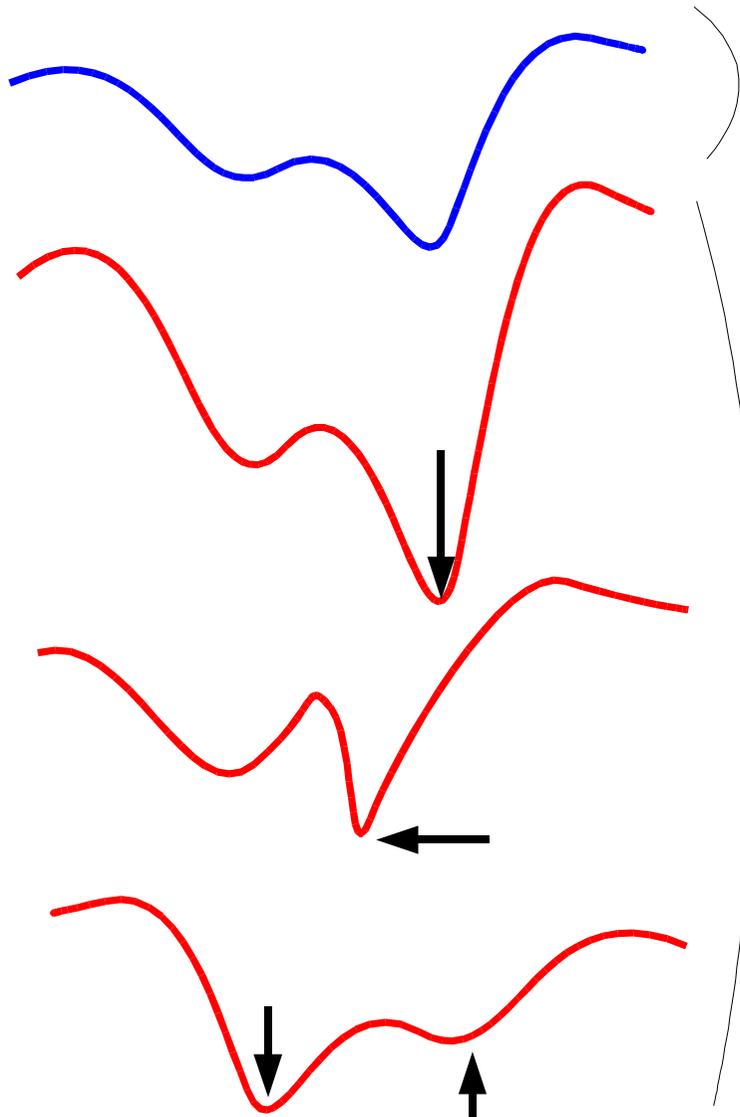
## Défauts :

- Nb. évaluations

Function	<i>CACO</i>			<i>API</i>		
	% ok	evals	err	% ok	evals	err
R2	100	6842	0.00		[10000]	0.00
SM	100	22050			[10000]	0.00
Gr5					[10000]	0.18

Function	<i>CIAC</i>			<i>HCIAC</i>		
	% ok	evals	err	% ok	evals	err
R2	100	11797	3e-3	100	18747	1e-8
SM	100	50000	9e-10	100	18616	5e-8
Gr5	63	48402	0.01	75	10870	1e-4

# Optimisation continue dynamique



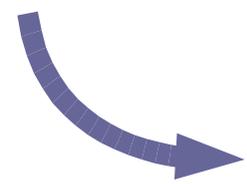
Formalisation :

- Classes de problèmes de base
- Benchmark

Mémoire

**Intensification** / « tracking »

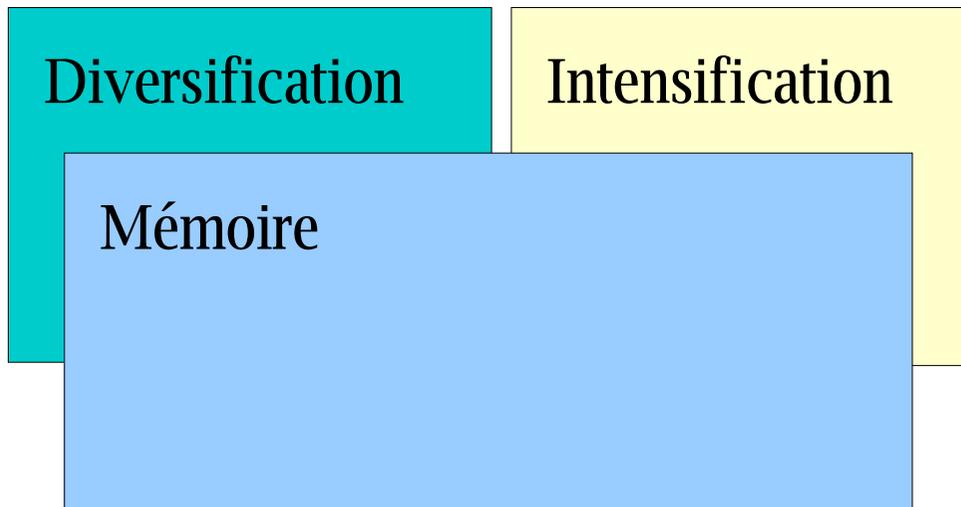
**Diversification** permanente



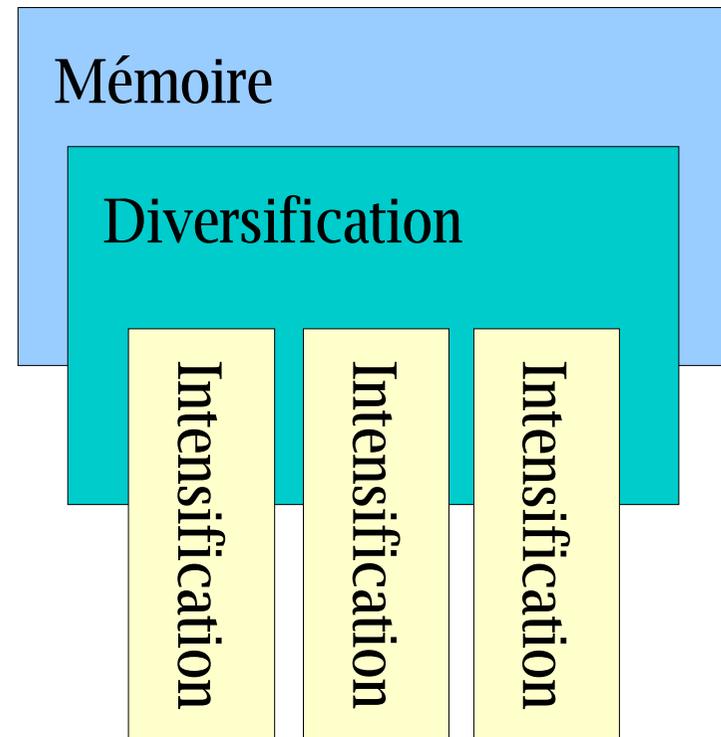
HCIAC

# Structure d'une métaheuristique

OEP



HCIAC



# Points importants

## BUT

Programmation  
à **mémoire** adaptative :

*Intensification*

*Diversification*

## MOYEN

Auto-organisation :

*Rétroactions*

*Flux d'infos*

Émergence



- **Population**
- **Parallélisme**
- **Flexibilité**



# Dréo, Pérowski, Siarry, Taillard

- Métaheuristiques
- Auto-organisation  
(colonies de fourmis)
- Programmation à  
mémoire adaptative

<http://nojhan.free.fr>