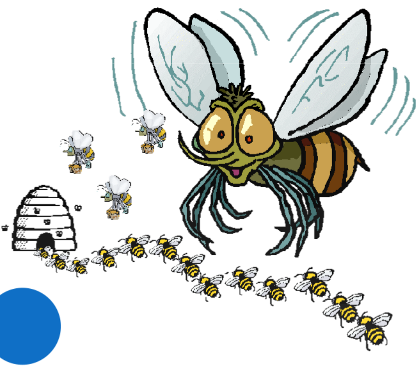
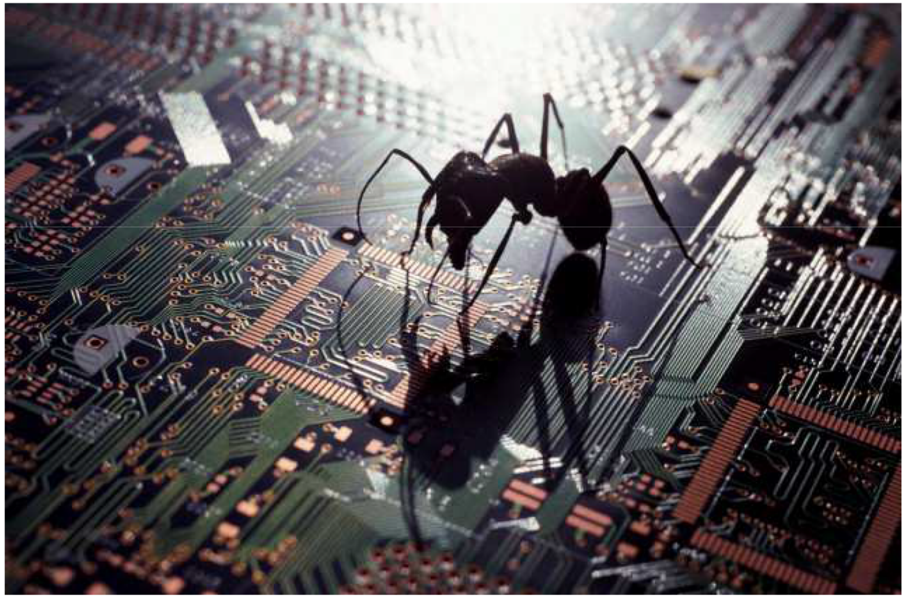
INERNATIONAL WORKSHOP ON OPTIMIZATION AND APPLICATIONS (WOA 2015)

LES MÉTAHEURISTIQUES : OUTILS DE CONCEPTION OPTIMALE DES CIRCUITS INTÈGRES ANALOGIQUES





Pr. Bachir BENHALA

FS - Meknès

ENSA – Kenitra le 23 Avril 2015

CONTEXTE

L’optimisation est un souci quotidien

- **Maximiser l’efficacité des moyens mis en œuvre par l’entreprise**

- **L’optimisation est la pierre angulaire des processus décisionnels**

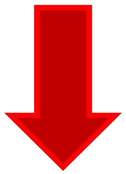
- **Complexité grandissante dans des domaines techniques**

  Télécommunications : Conception de réseaux mobiles UMTS
  Transport : Problèmes de tournées de véhicules
  Électronique : Placement et routage des composants

Problématique

Le besoin d'efficacité et de rentabilité toujours croissant

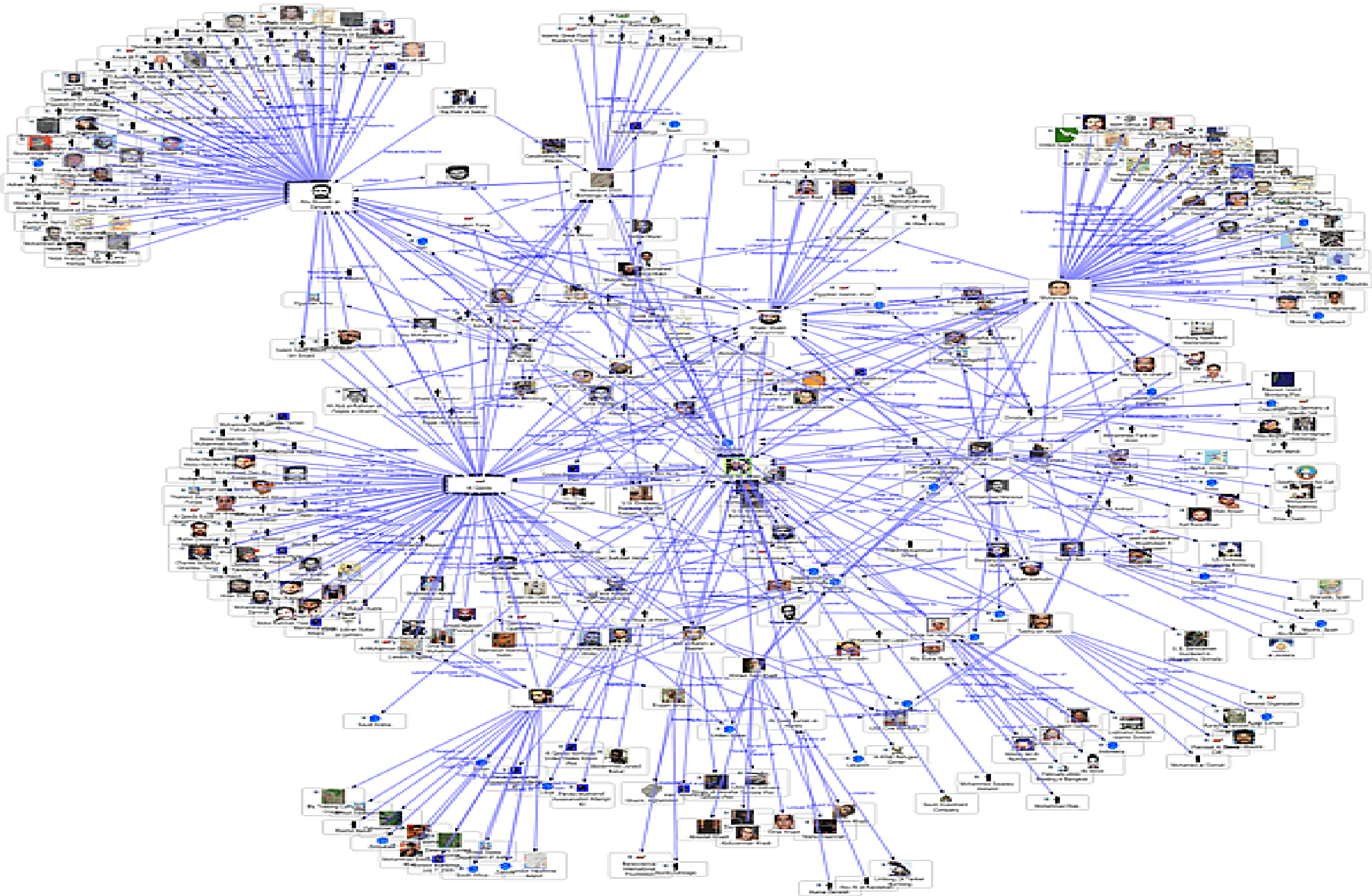
Ne plus se référer seulement à l'expérience des experts



Résultats numériques obtenus par la résolution de problèmes d'optimisation

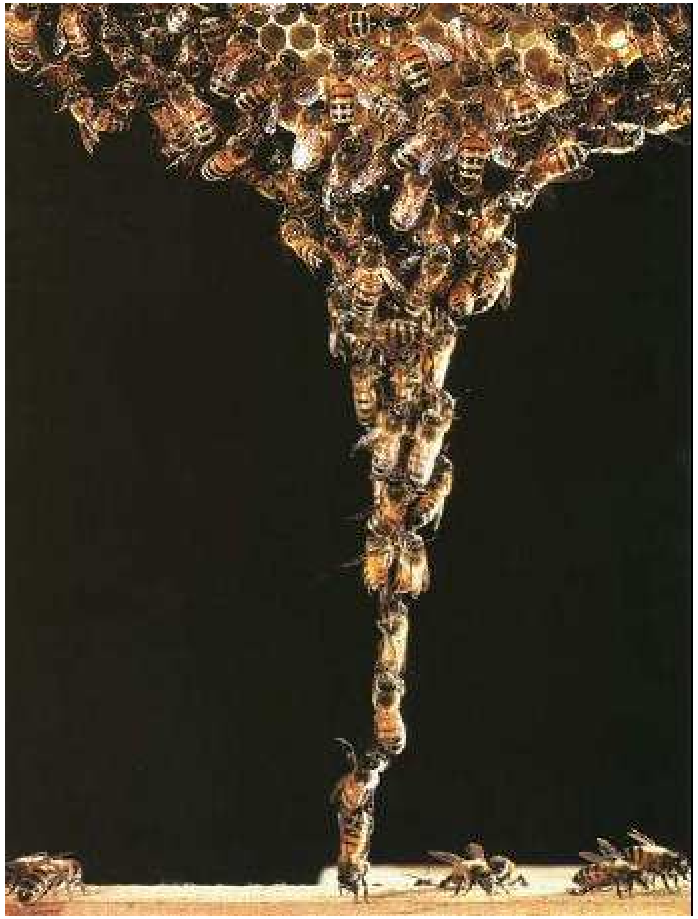
PROBLÉMATIQUE

Les problèmes d'aujourd'hui sont de plus en plus complexe alors que les ressources pour les résoudre sont limitées



SOLUTION

La nature fournit des mécanismes hautement évolués



SOMMAIRE

1. **Problème d'optimisation**
2. **Les métaheuristiques**
3. **Evaluation des performances**
4. **Parallélisme**
5. **Hybridation**
6. **Application**
7. **Conclusion**

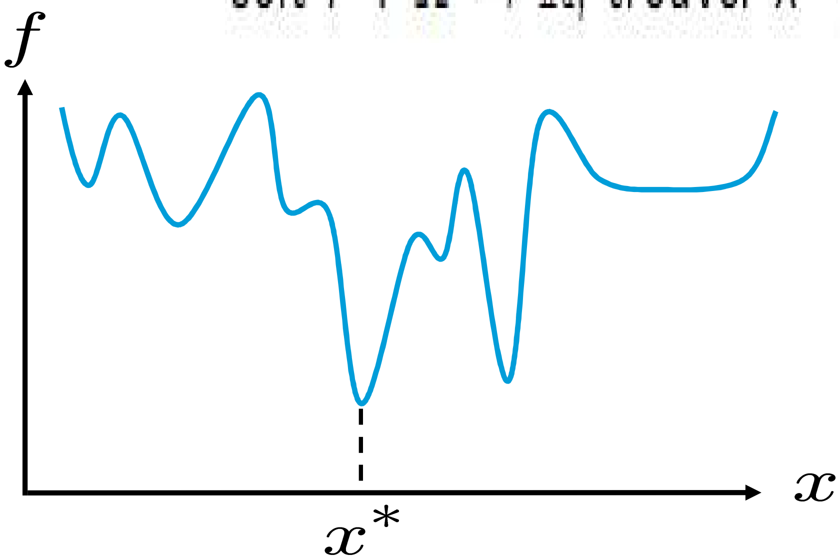
SOMMAIRE

1. **Problème d'optimisation**
2. Les métaheuristiques
3. Evaluation des performances
4. Parallélisme
5. Hybridation
6. Application
7. Conclusion

1. PROBLÈME D'OPTIMISATION | PRINCIPE

- **Optimiser : permettre d'obtenir le meilleur résultat possible par une action bien adaptée.**
- **Synonymes : améliorer, maximiser, mettre au point.**

soit $f : \Omega \mapsto \mathbb{R}$, trouver $x^* \in \Omega$ tel que $f(x*) = min_{x \in \Omega} f(x)$



$\Omega\,(ou\ S)$: espace de recherche

f : fonction objectif

$x^*\,(ou\ s^*)$: optimum, minimum

Les problèmes d'optimisation sont toujours ramenés à une minimisation de la fonction objectif:

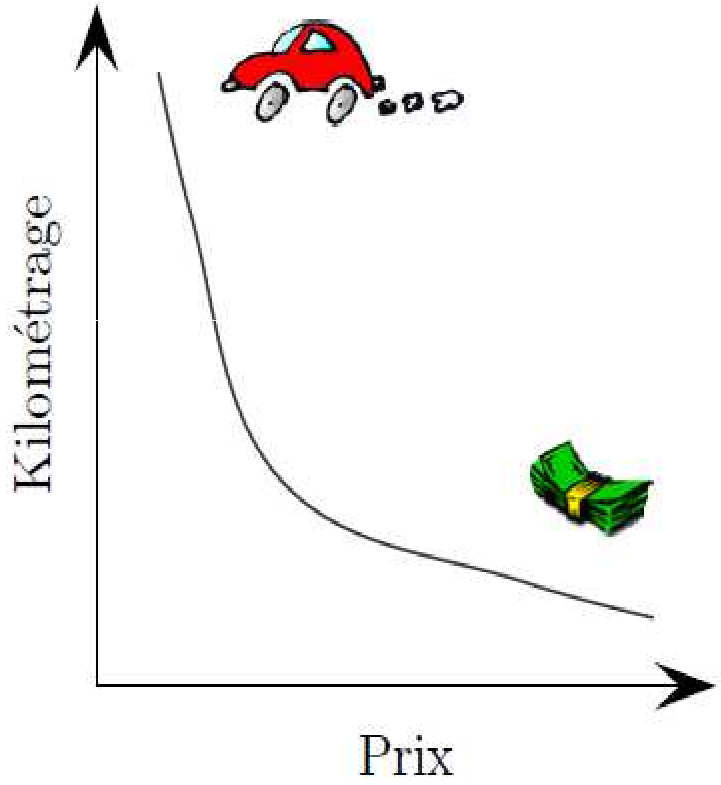
$$\text{maximiser } f(\vec{x}) \Longleftrightarrow \text{ minimiser } -f(\vec{x})$$

1. Problème d'optimisation | Principe

- **Les problèmes réels mettent en fait en jeu plusieurs objectifs simultanément**

- **Le plus souvent contradictoires :**
  coût et qualité

- **Pas une seule solution :**
  mais des compromis



Multi-objectifs

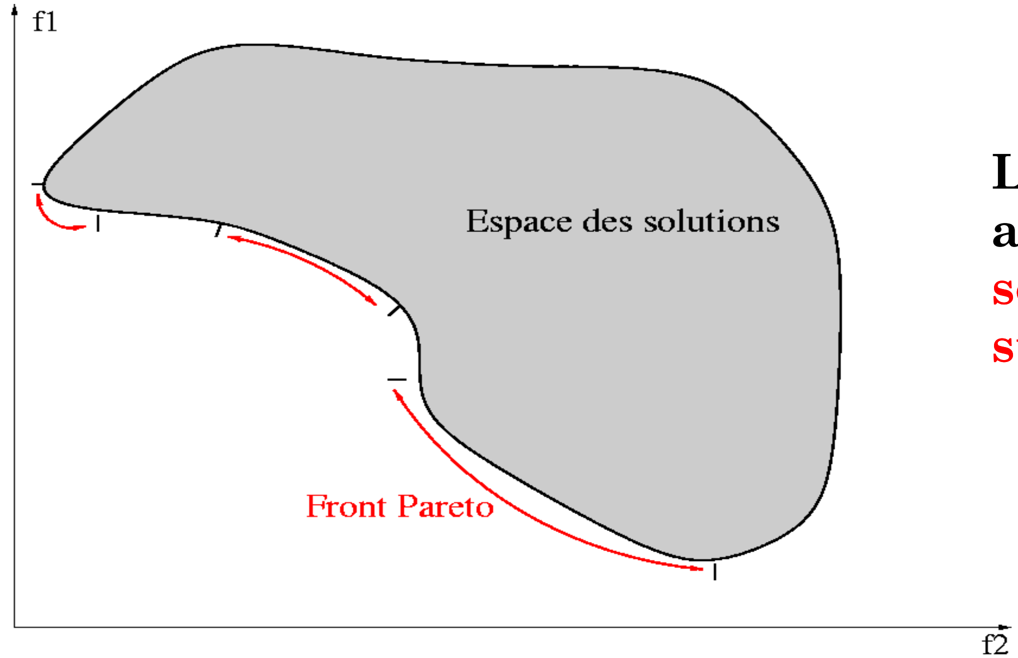
<u>But :</u> Optimiser n fonctions objectifs

$$Minimiser\ \vec{f}\,(\vec{x}) = \left(f_1(\vec{x}), f_2(\vec{x}), \cdots, f_k(\vec{x})\right)$$

$$g_i(\vec{x}) \leq 0, \quad i = 1,2,\cdots,m$$

$$h_i(\vec{x}) = 0, \quad i = 1,2,\cdots,p$$

<u>Résultat :</u>
Ensemble de solutions optimales (**Front Pareto**)

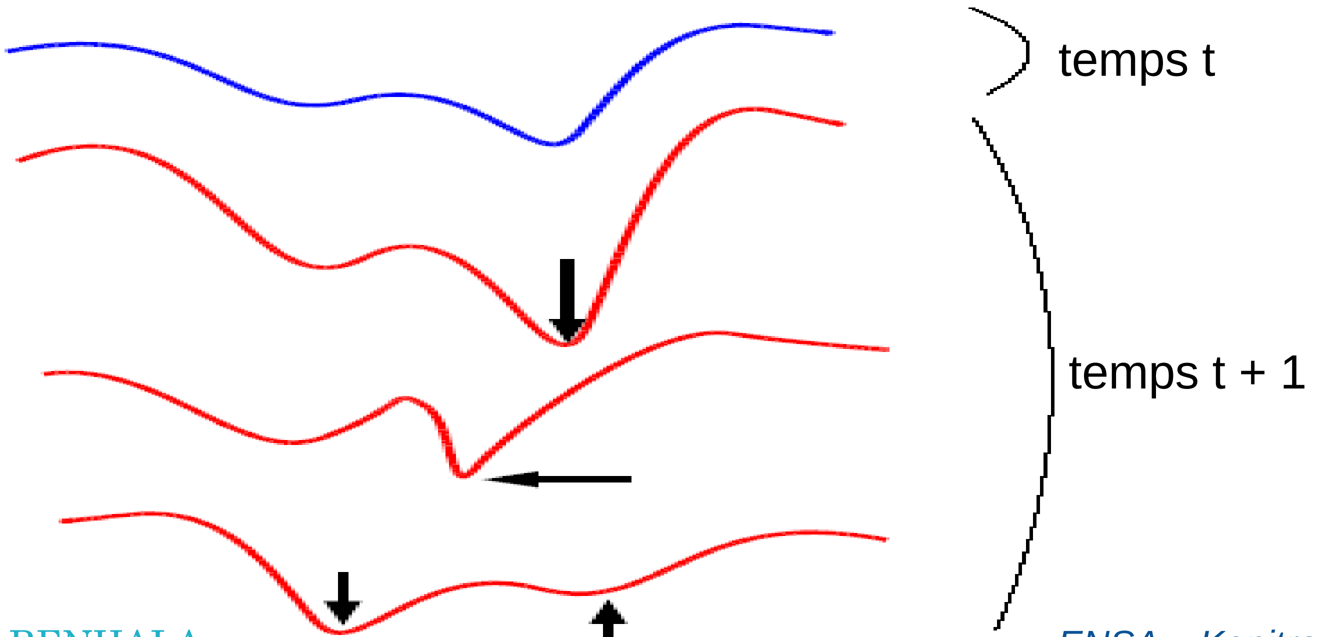


Le front Pareto est aussi appelé l'ensemble des solutions efficaces ou la surface de compromis

- **<u>Optimisation combinatoire</u> consiste en l'optimisation d'un certain critère sous différentes contraintes**

- **<u>Optimisation multimodale</u> où l'on ne cherche plus l'optimum global, mais l'ensemble des meilleurs optima locaux**

- **<u>Optimisation dynamique</u> où il faut approcher l'optimum a chaque pas de temps, car la fonction objectif change de topologie au cours du temps**



Problème du voyageur de commerce
Traveling Salesman Problem (TSP)

Trouver le court chemin pour traverser un ensemble de villes en ne traversant jamais la même ville deux fois

Nombre de villes	Nombre de possibilités	Temps de calcul
5	12	12 nanosecondes
10	181 440	0.18 millisecondes
15	43 milliards	43 secondes
20	60×10^{15}	23.14 mois
25	310×10^{21}	9.8 millions d'années (!)

TSP pour 120 villes en Allemagne

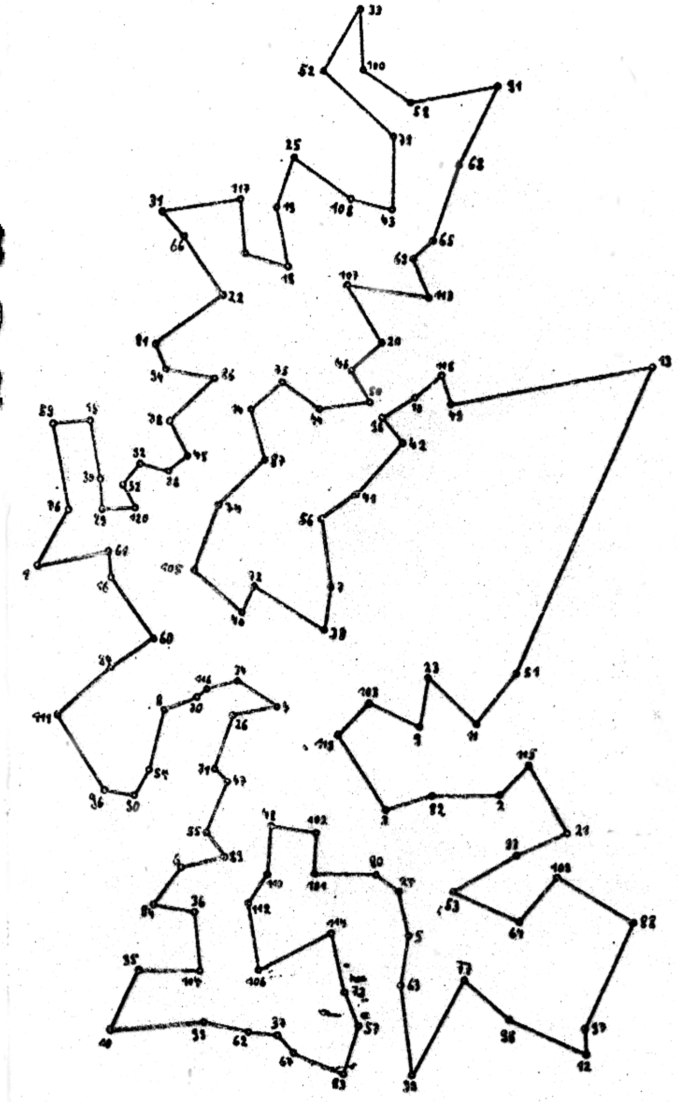
Nombre de solutions possibles =

2679254907606348937555461899482198739957886903776 8
7078048465194329577247030862734015632117088075939 9
8691345929648364341894253344564803682882554188736 2
42799920969079258554704177287.

(179 chiffres)

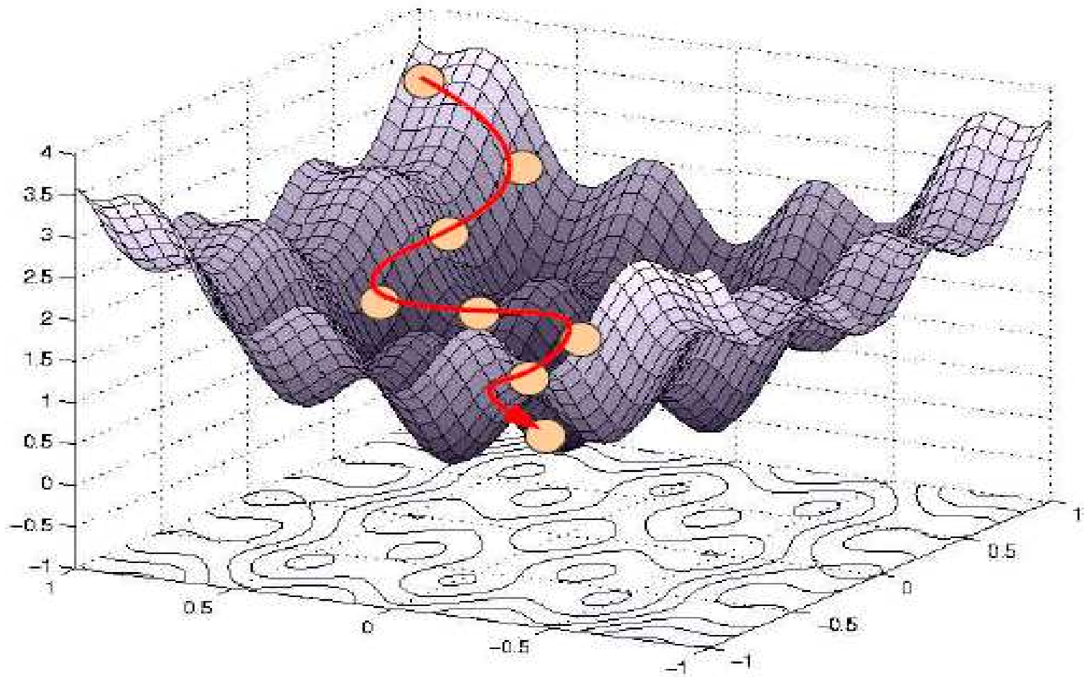
Métaheuristiques

La tournée minimale est de 6942 km



1. Problème d'optimisation : Solution

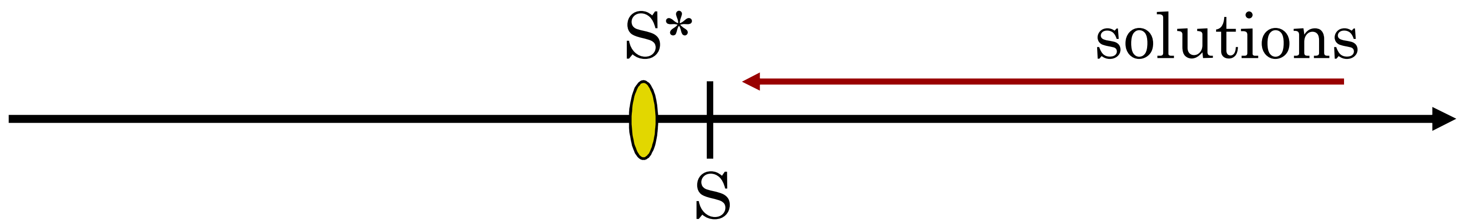
Les métaheuristiques sont des algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficile pour lesquels les méthodes classiques ne sont pas applicables



SOMMAIRE

1. Problème d'optimisation
2. **Les métaheuristiques**
3. Evaluation des performances
4. Parallélisme
5. Hybridation
6. Application
7. Conclusion

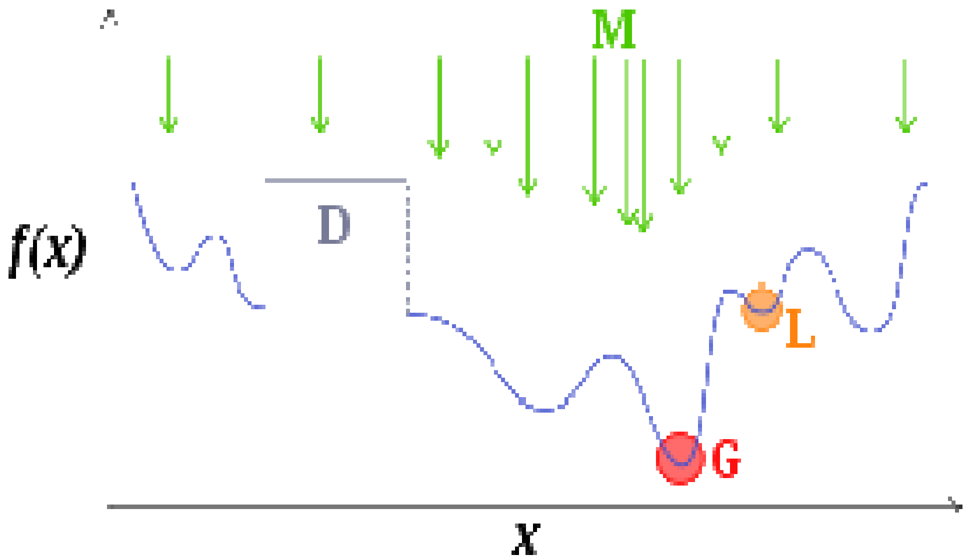
Trouvent, ou convergent, vers l'optimum global



Métaheuristiques

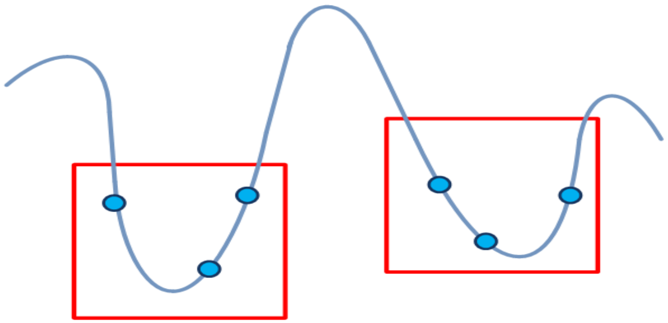
Très utilisées pour la résolution des problèmes NP-difficile

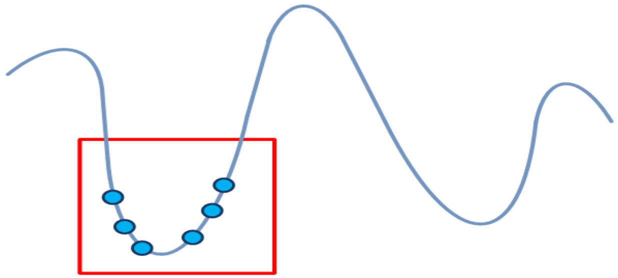
Les métaheuristiques progressent vers un optimum global par échantillonnage de la fonction objectif



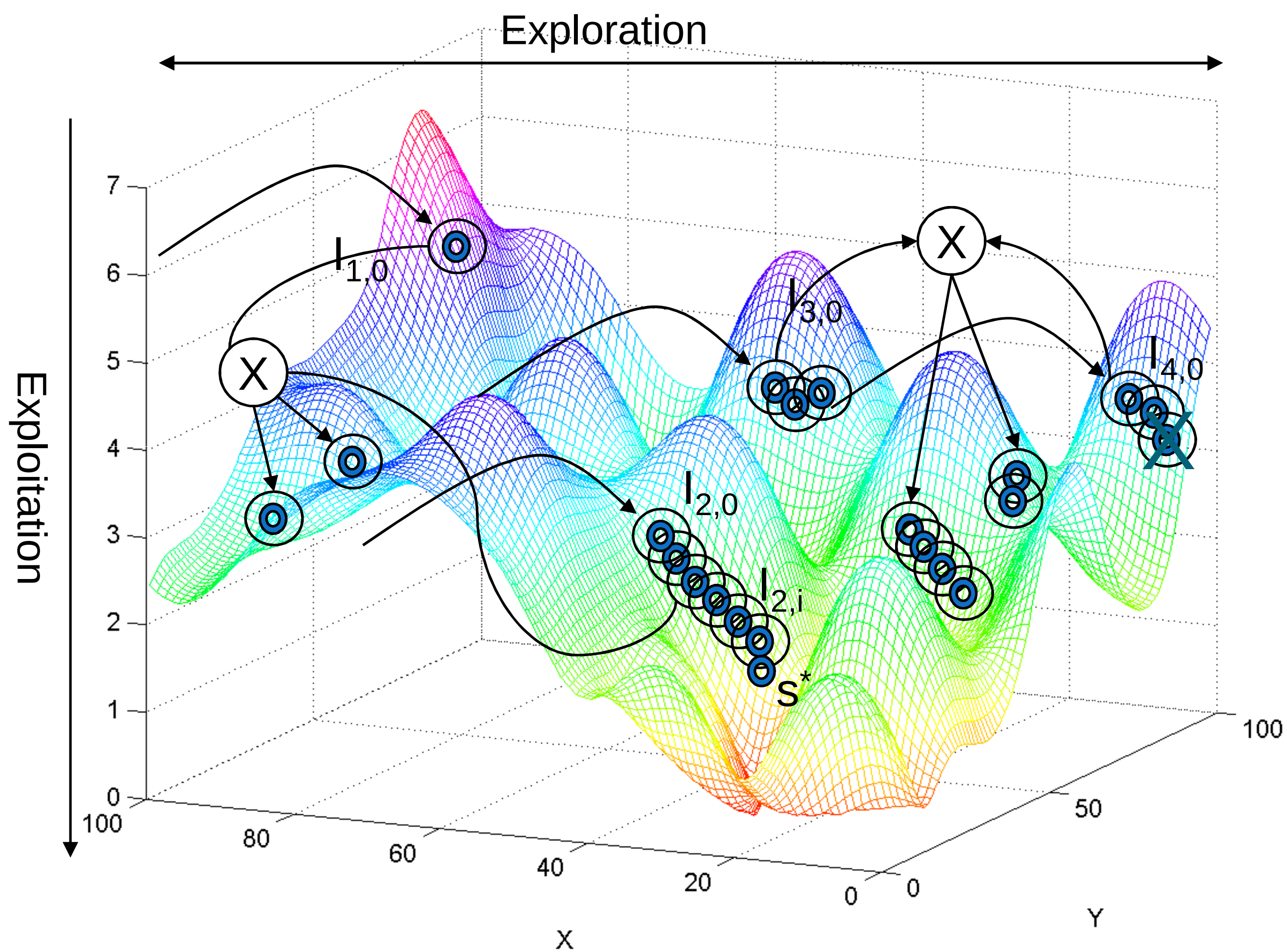
Diversification « Exploration »

Intensification « Exploitation »





2. Les Métaheuristiques | Principe



Avantages

1. Simple à implémenter
2. Rapidité de résultats
3. Solutions de bonne qualité
4. Convenable pour un nombre large de problèmes

Inconvénients

1. Demandent un bon paramétrage
2. Pas de garantie de l’optimum global

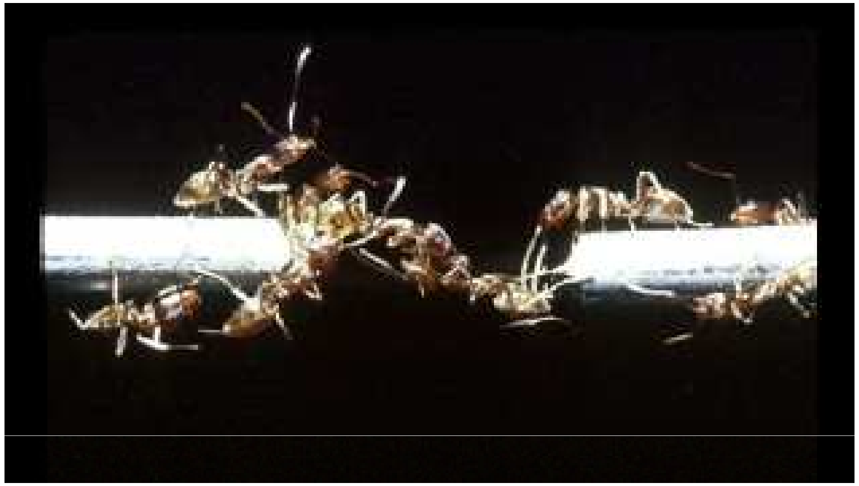
o Algorithme à recherche locale (voisinage)

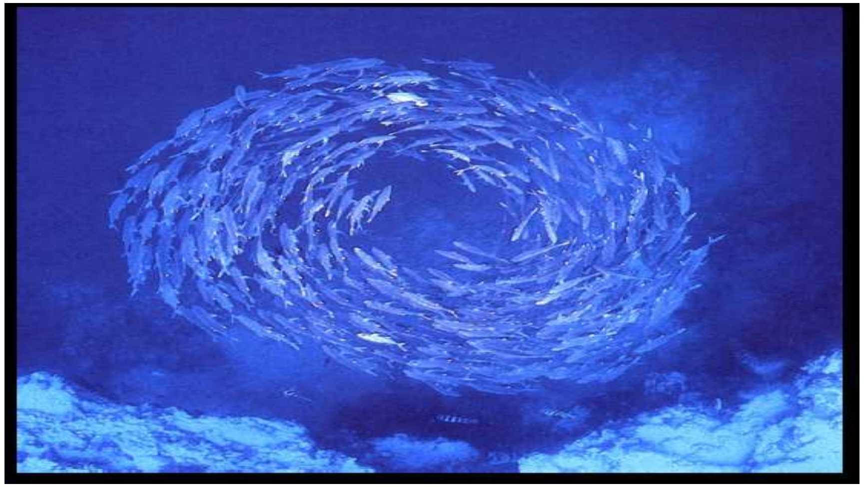
- ◊ Recuit simulé
- ◊ Recherche tabou
- ◊ Méthodes de descente
- ◊ Algorithme glouton
- ◊…

o Algorithme à population

- ◊Algorithme génétique
- ◊Algorithme des colonies de fourmis
- ◊Essaims particulaires
- ◊Algorithmes des abeilles
- ◊Algorithme des chauve-souris
- ◊Nid de guêpes
- ◊la recherche de nourriture bactérienne
- ◊…

L'intelligence collective

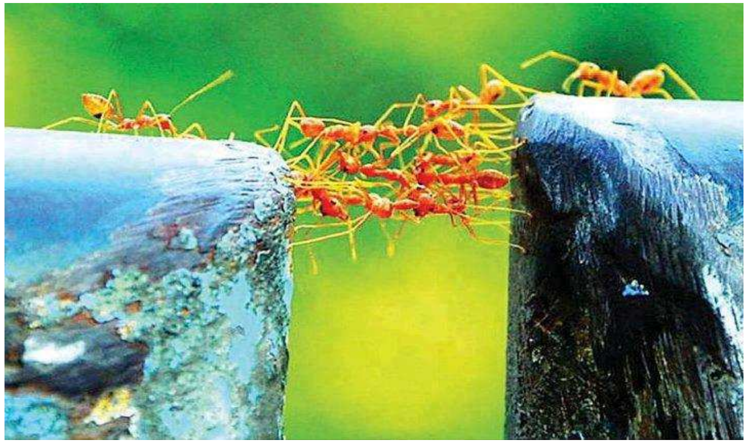




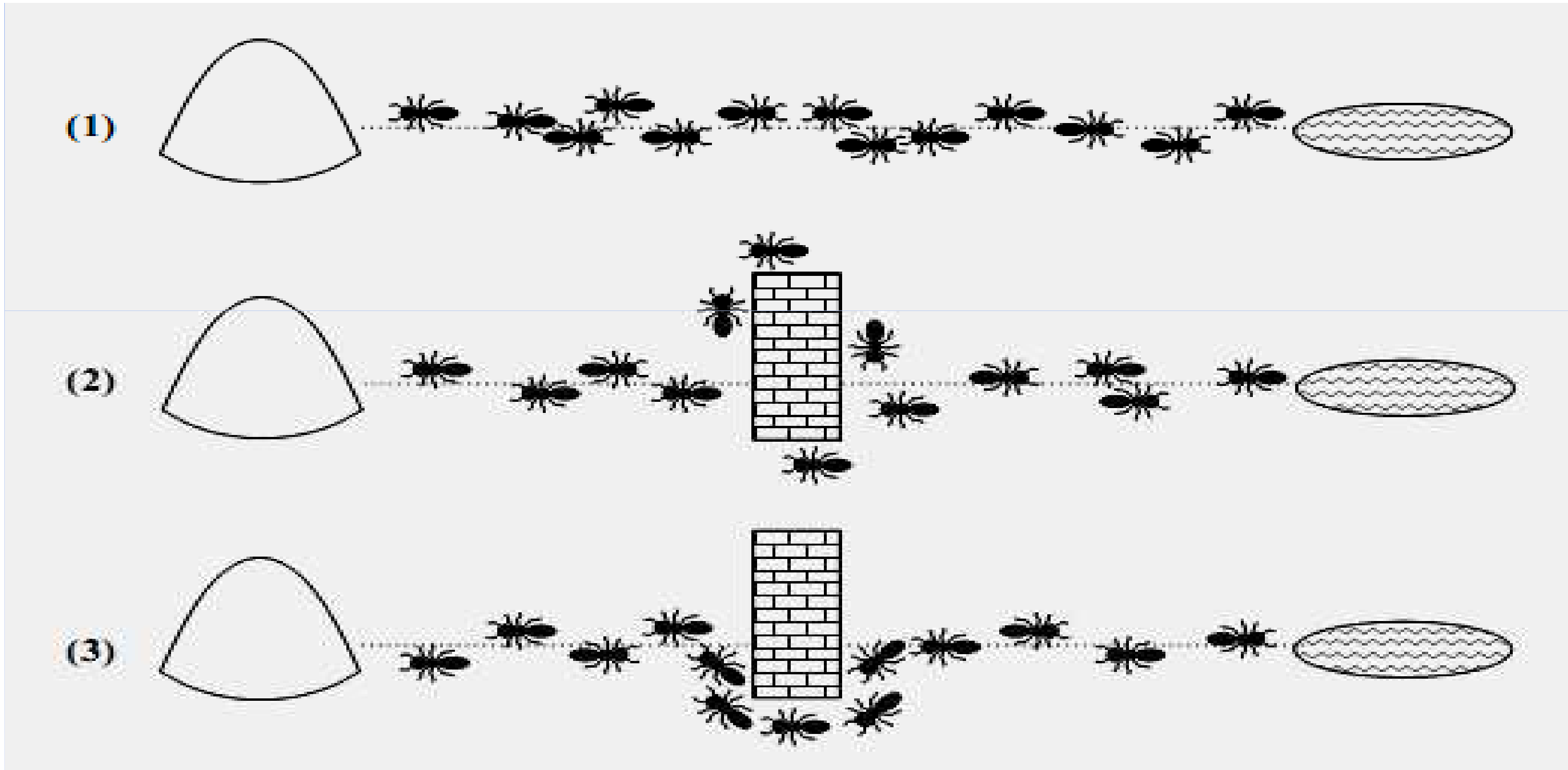
- Composés d'agents *simples*
- Décentralisés
- Comportement émergent
- Robustes et flexibles

Ant colony Optimization (ACO)1996, Dorigo

Cette métaheuristique est inspirée des comportements collectifs de fourmis



Capacité des fourmis à retrouver le chemin le plus court



Algorithme des colonies de fourmis : Formulation

Le déplacement d'une fourmi *k* d'un point *i* a un point *j* se fait suivant la probabilité:

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{\left(\tau_{ij}(t)\right)^{\alpha} \cdot (\eta_{ij})^{\beta}}{\sum_{l \in J_i^k} \left(\tau_{il}(t)\right)^{\alpha} \cdot (\eta_{il})^{\beta}} & Si \quad j \in J_i^k \\ 0 & Si \quad j \notin J_i^k \end{cases}$$

Chaque fourmi laisse une certaine quantité de phéromone sur l'ensemble de son parcours

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L^k(t)} & Si\ (i,j) \in T^k(t) \\ 0 & Si\ (i,j) \notin T^k(t) \end{cases}$$

La règle de mise à jour des pistes:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t)$$

Algorithme des colonies de fourmis : Algorithme

Initialisation aléatoire des phéromones pour chaque fourmi

Pour $t = 1, \cdots, t_{max}$

  Pour chaque fourmi $k = 1, \cdots, m$ **faire**

    Choisir une ville au hasard

    Pour chaque ville non visitée i **faire**

      Choisir une ville j, dans la liste J_i^k des villes restantes

    Fin Pour

    Déposer une piste de phéromone $\Delta\tau_{ij}^k(t)$ sur le trajet $T^k(t)$

  Fin Pour

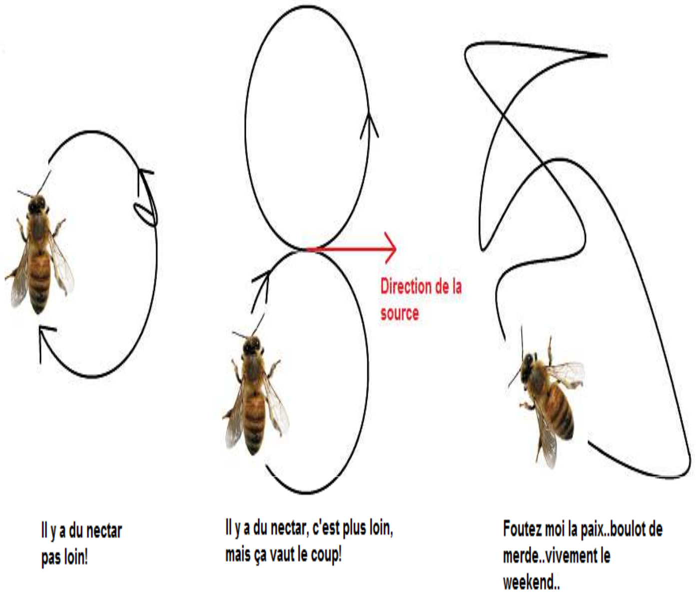
  Evaporer les pistes

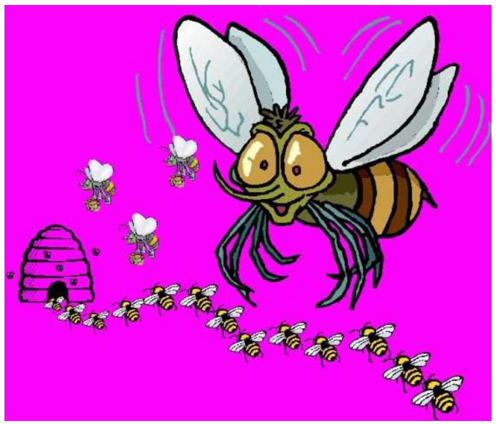
Fin Pour

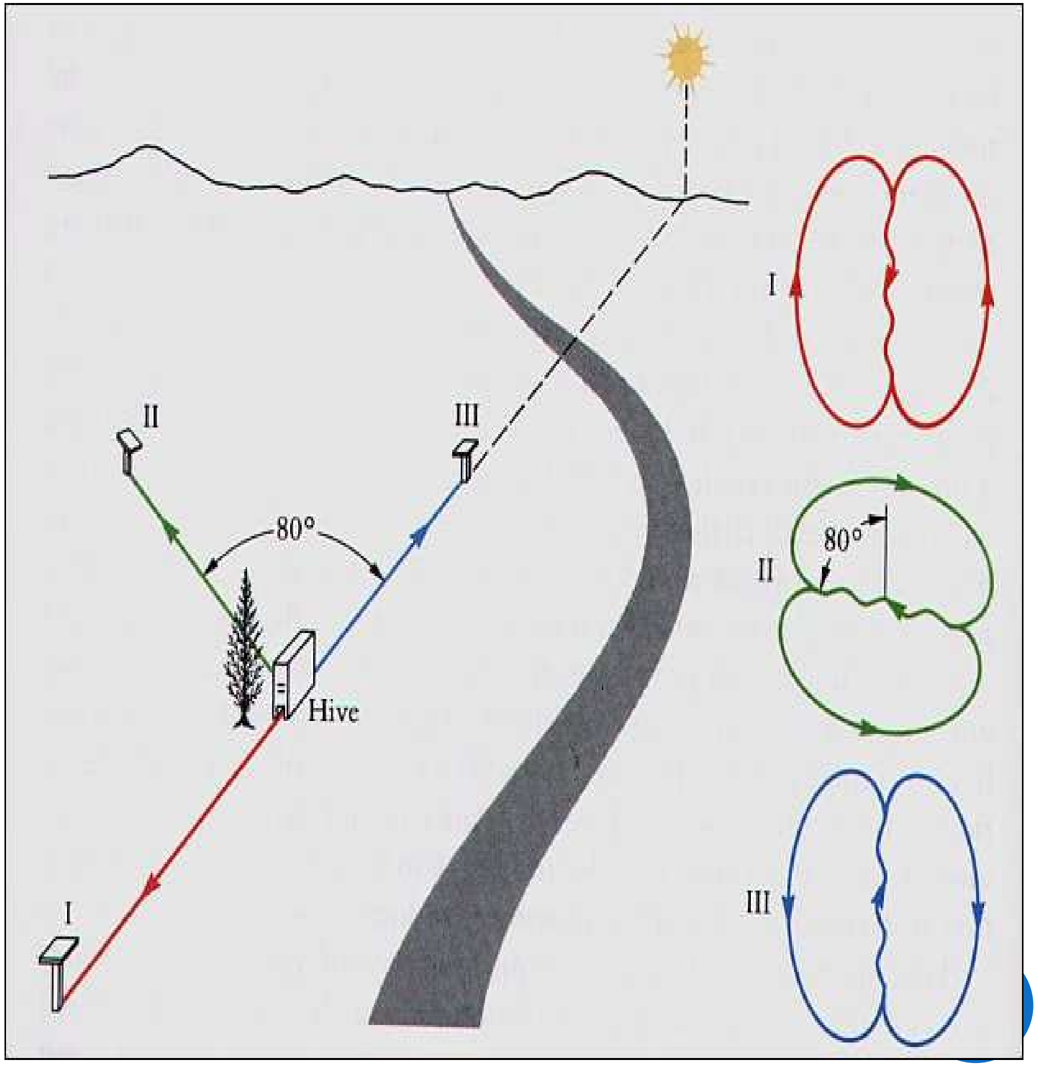
Artificial bee colony algorithm (ABC)

2006, B.BASTURK, D.JARABOGO

Algorithme des abeilles







Algorithme des abeilles

- Une méthode d'optimisation inspirée du comportement intelligent de l'abeille lors de la récolte du nectar.
- Abeille: insecte sociale, très organisée.
- Trois types d'abeilles:

  ouvrière, faux-bourdons, une seule reine.
- Trois types de butineuses:

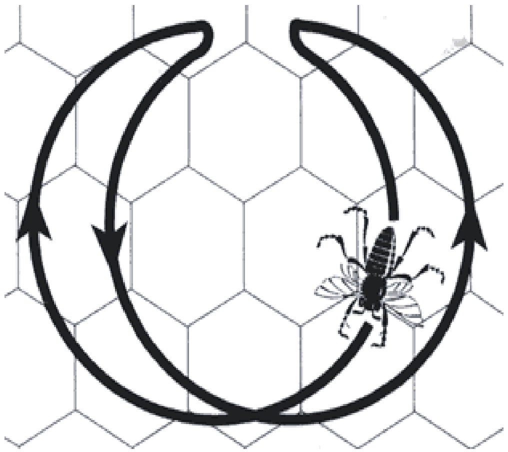
  active, inactives, éclaireuse.

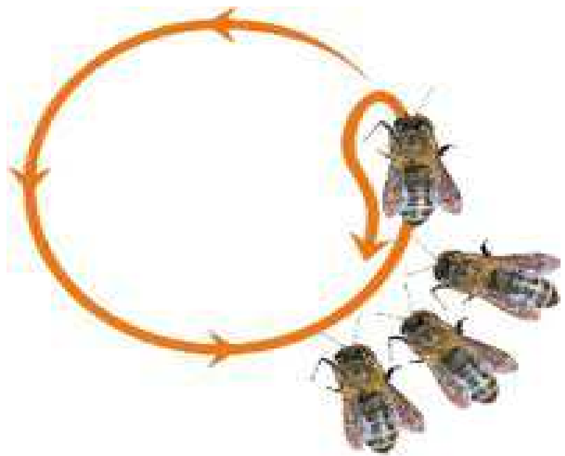
But : système multi-agents. Processus de résolution réalisé par l'interaction entre ces agents.

Comportement des abeilles

- Principe de base : coopération entre les abeilles, faite par la communication via une danse, pour transmettre des informations sur les sources de nourriture.
- Deux types de danse :

  - Danse en rond => pollen à faible distance.
  - Danse frétillante => à moins de 10 Km.





Algorithme des abeilles

- Initialiser la population avec n solutions aléatoires
- Evaluer la fitness de la population.

- **Tant que** *le critère d'arrêt n'est pas satisfait* **faire**

Recruter des abeilles -> rechercher de nouvelle source de nourriture.

- Evaluer la fitness de la population.

Si *un membre de la population ne s'est pas amélioré*

Faire enregistrer la solution et remplacer la par une solution aléatoire.

Trouver les solutions aléatoires et remplacer les membres de la population qui ont la mauvaise fitness.

- **Fin Tant que.**
- **Retourner** *la meilleure solution.*

SOMMAIRE

Benchmark de test

- Valider l'algorithme établi
- Paramétrer l'algorithme

- Il présente un difficulté particulière pour la convergence ou pour la diversité

- La forme et la position de la surface de Pareto soient connues et les valeurs des variables de décision correspondantes soient faciles à trouver

Michalewicz (MZ)

(n variables)

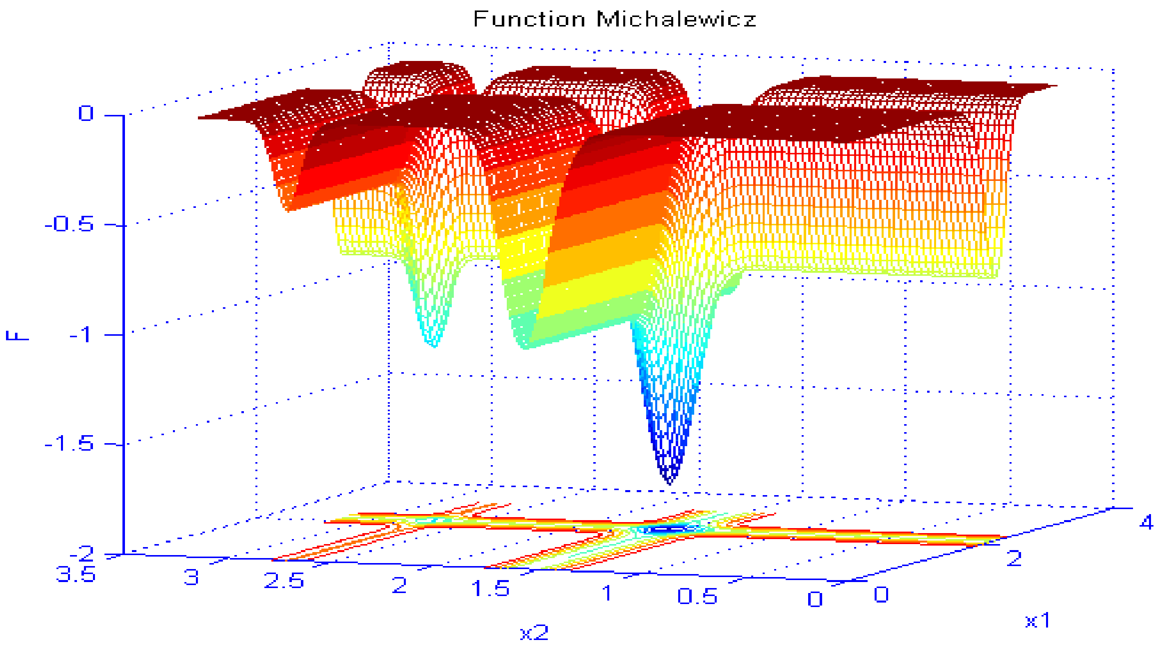
$$MZ(x) = \sum_{i=1}^{n} \sin(x_i) . \left[\sin\left(\frac{i . x_i^{2}}{\Pi} \right) \right]^{20}$$

domaine de recherche : $0 \leq x_i \leq \pi$, i= 1, … n

n = 2 , 1 minimum global : MZ(x*) = - 1.80

n = 5 , 1 minimum global : MZ(x*) = - 4.687

n =10, 1 minimum global : MZ(x*) = - 9.68

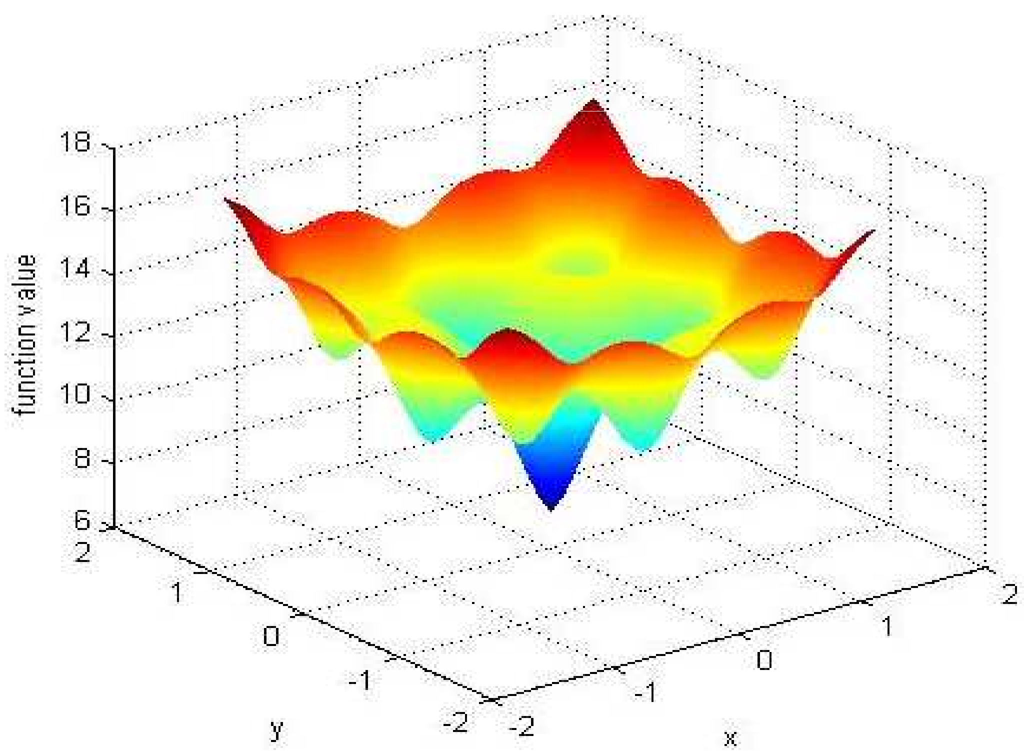


Ackley's Function

$$f(x,y) = \frac{1}{f}\left\{-a\exp\left[-b\sqrt{\frac{1}{n}(x^2+y^2)}\right] - \exp\left[\frac{1}{n}(\cos(cx)+\cos(cy))\right] + a + \exp(1) + d\right\},$$

where $a = 20,\ b=0.2,\ c=2\pi,\ d=5.7,\ f=0.8,\ n=2,$

$-1.5 \leq x \leq 1.5$ and $-1.5 \leq y \leq 1.5.$



(0, 0)

(-0.9522, 0)

(0.9522, 0)

(0, -0.9522)

(0, 0.9522)

(-0.9685, -0.9685)

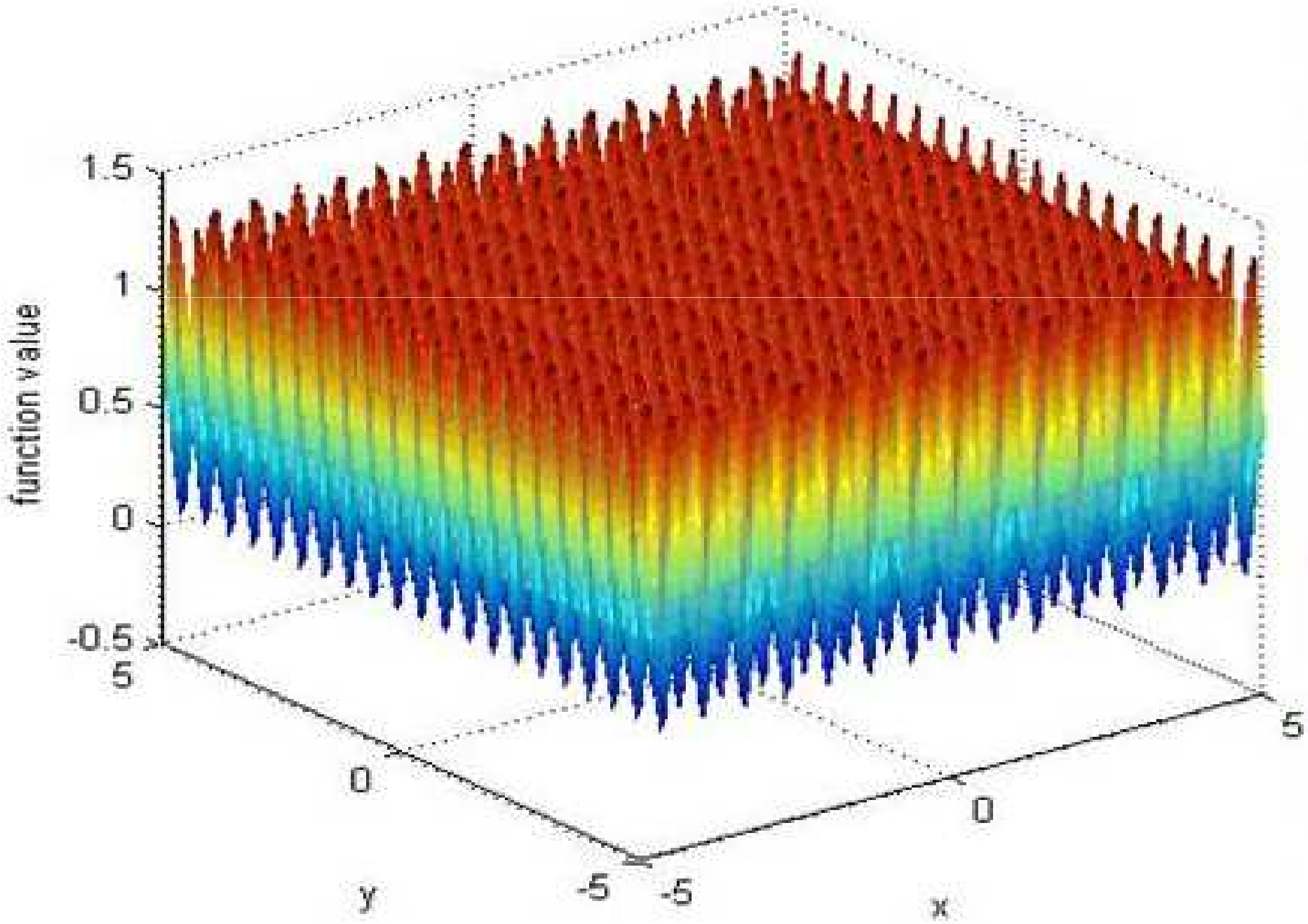
(0.9685, 0.9685)

(0.9685, -0.9685)

(-0.9685, 0.9685)

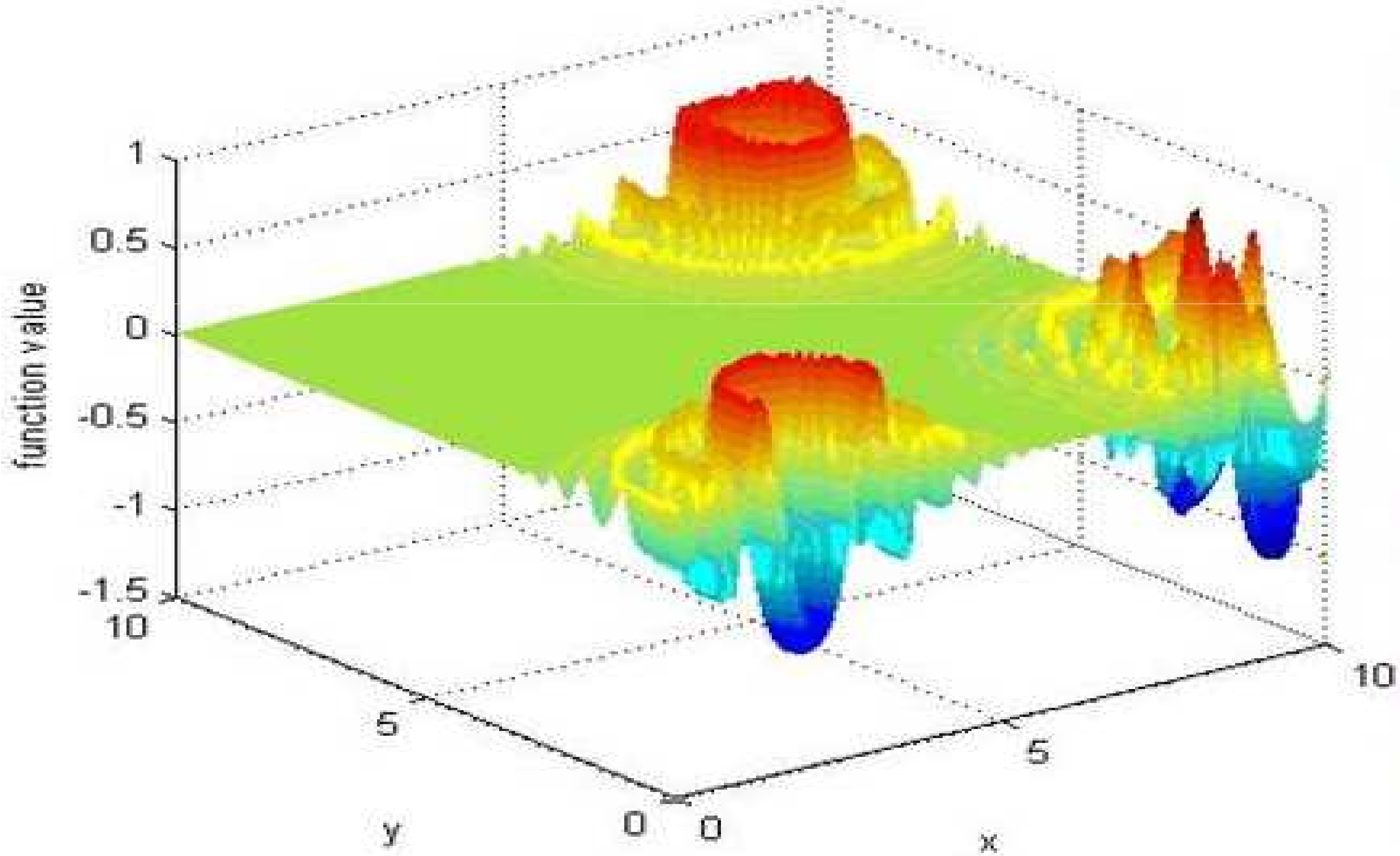
Bohachecsky

$$\min_{x} f(x) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7$$



Modified Langerman

$$\min_x f(x) = -\sum_{j=1}^{5} c_j \cos(d_j/\pi) \exp(-\pi d_j)$$



Multi-objectifs

Le benchmark **ZDT** tient son appellation des initiales des trois chercheurs qui l'ont proposé, en l'occurrence Zitzler, Deb et Thiele.

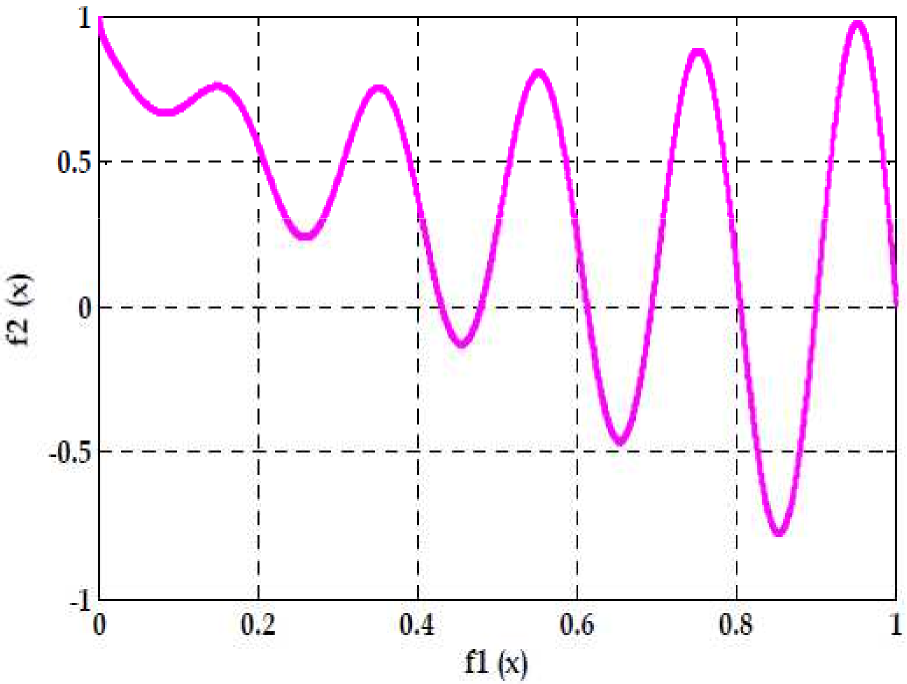
ZDT3

$$f_1(\vec{x}) = x_1$$

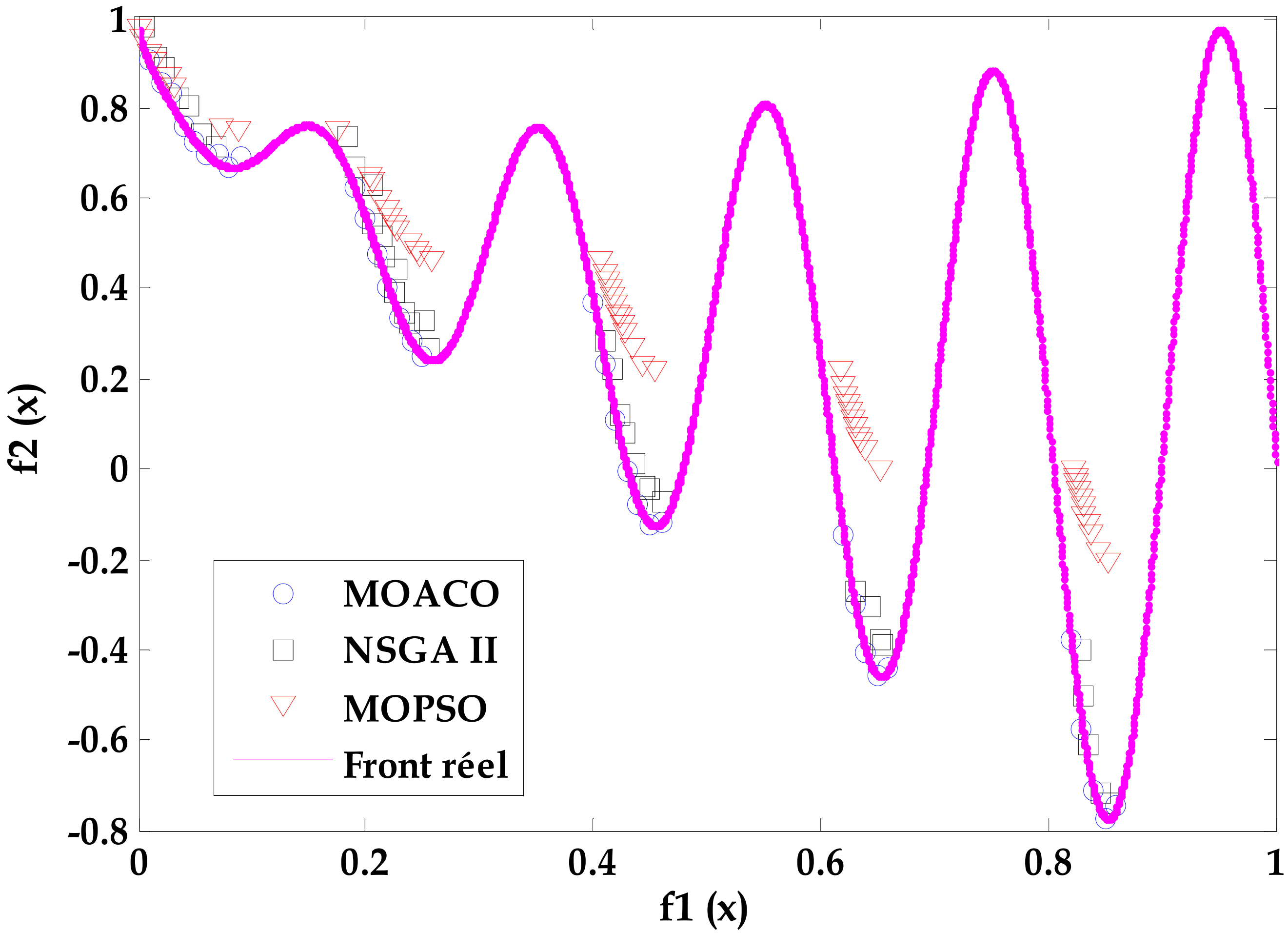
$$g(\vec{x}) = 1 + 9\sum_{i=2}^{D} \frac{x_i}{D-1}$$

$$h(\vec{x}, f_1, g) = 1 - \left(\sqrt{\frac{f_1(\vec{x})}{g(\vec{x})}}\right) sin\left(10\pi f_1(\vec{x})\right) - \sqrt{\frac{f_1(\vec{x})}{g(\vec{x})}}$$

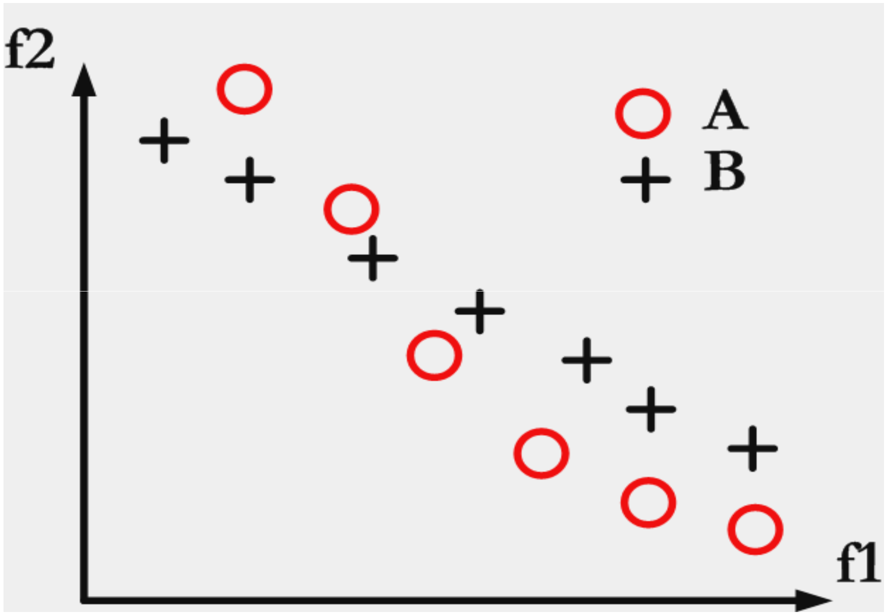
$$f_2(\vec{x}) = g(\vec{x}) \cdot h(\vec{x}, f_1, g)$$



ZDT 3 pour n=30

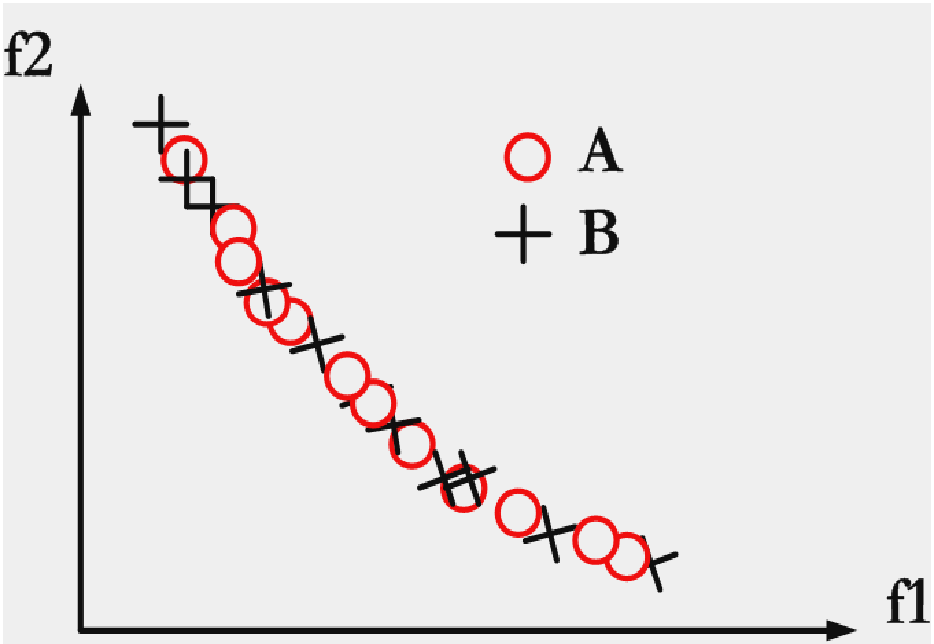


La métrique Covariance

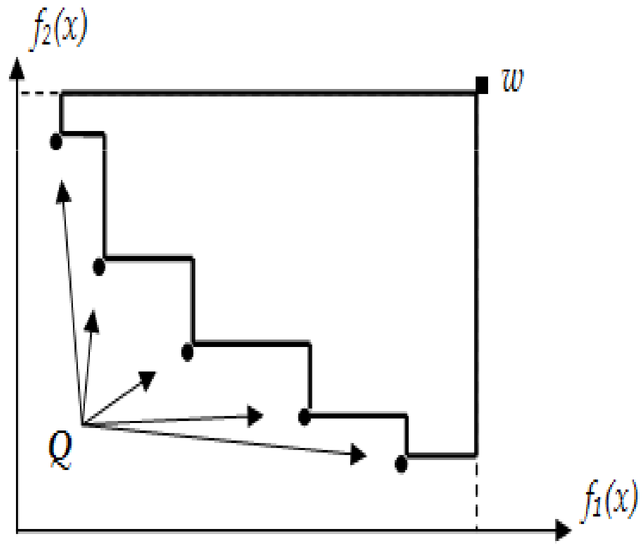


$$C(A,B) := \frac{|\{b \in B \mid \exists a \in A : a \geq b\}|}{|B|}$$

La métrique Hypervolume



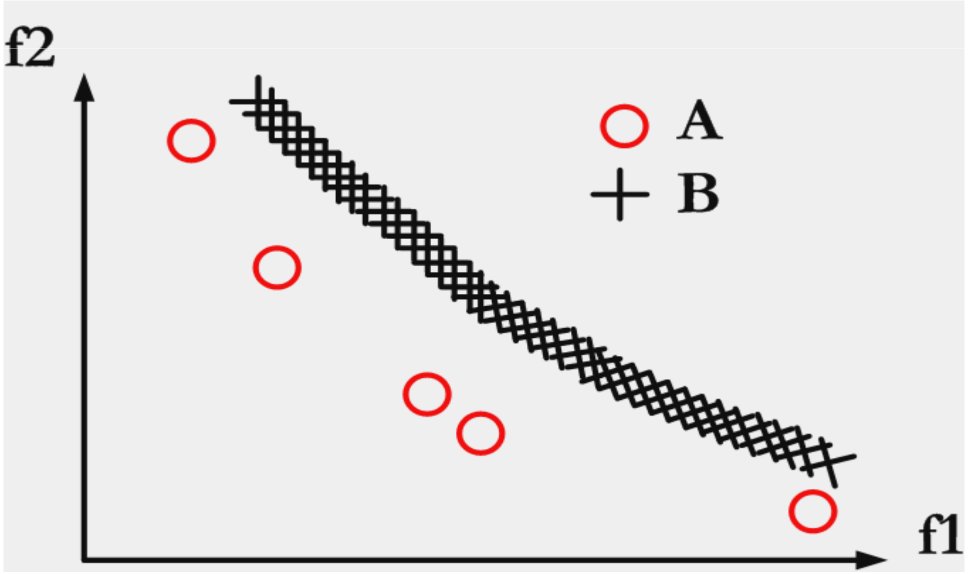
$$HV = volume\left(U_{i=1}^{|Q|} C_i\right)$$



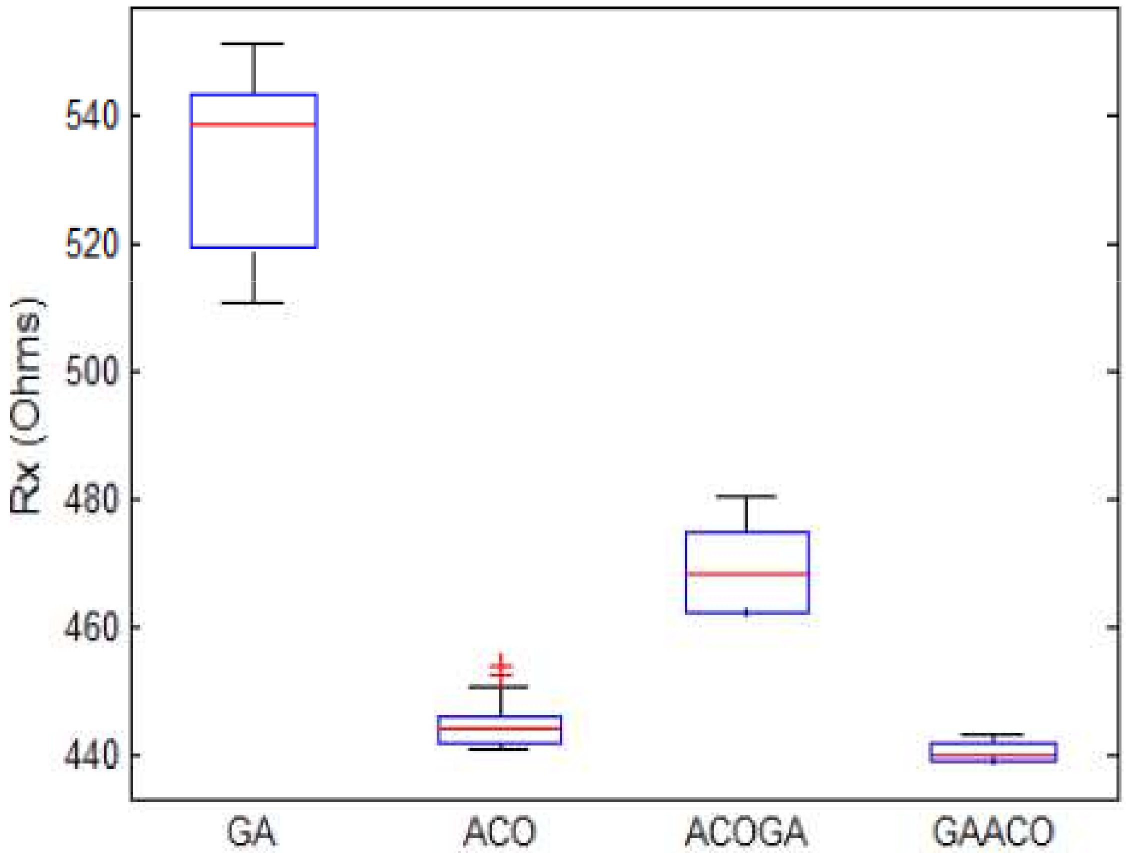
La métrique Espacement

$$S(A) = \sqrt{\frac{1}{A-1}\sum_{i=1}^{|A|}\left(d_i - d_{moy}\right)^2}$$

$$d_i = \min_{s_j \in A \wedge s_j \neq s_i} \left(\sum_{u=1}^{k}\left|f_u(s_i) - f_u(s_j)\right|\right)$$



vérifier le taux de convergence de la métaheuristique vers le même optimum



Algorithmes	Rapidité	optimalité	robustesse	Mise en œuvre
ACO	- -	+ +	+ +	difficile+
PSO	+ -	+ +	+ -	difficile
ABC	+ -	+ -	+ -	difficile+
GA	+ +	+ -	+ -	simple
SA	+ +	+ -	- -	simple

Les métaheuristiques à base de population nécessitent beaucoup de temps pour les instances de grande taille

Sommaire

4. PARALLÉLISME

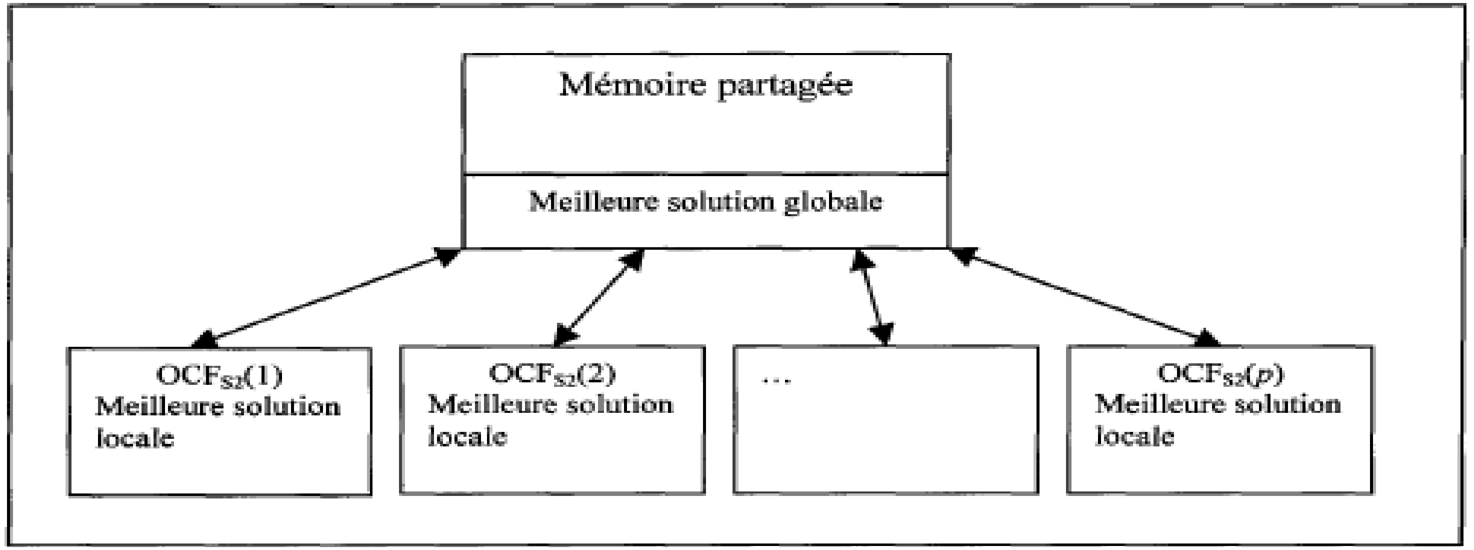
La structure même des métaheuristiques à population comporte un parallélisme intrinsèque.

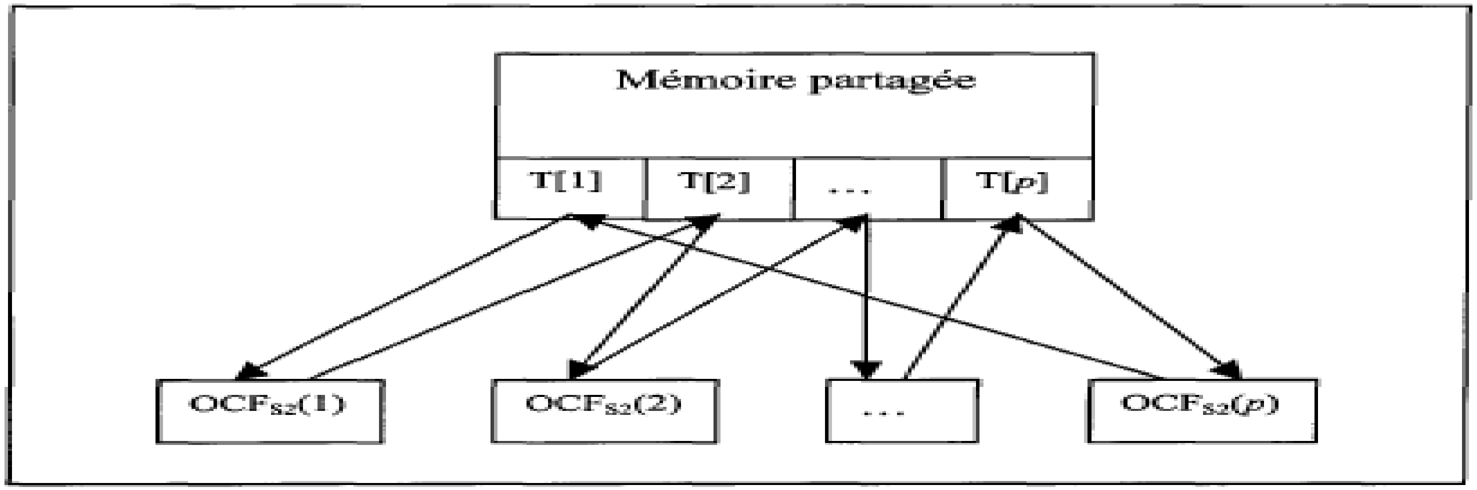
Les métaheuristiques à population sont les mieux adaptées au parallélisme par des systèmes multi-agents ou multi-population

Nécessité d'architecture parallèle

Machine à multiprocesseur

4. PARALLÉLISME





Sommaire

1. Problème d'optimisation
2. Les métaheuristiques
3. Evaluation des performances
4. Parallélisme
5. **Hybridation**
6. Application
7. Conclusion

Objectif

Profiter de la puissance respective de chacune des métaheuristiques

Trois types d'hybridation :

- Niveau « Simple »
- Niveau « Evolué »
- Niveau « Coopératif»

5. HYBRIDATION | « NIVEAU SIMPLE »

INITIALISATION
de paramètres

↓

DIVERSIFICATION
détection de « zones prometteuses »
$\Rightarrow$ liste prometteuse

1ère Métaheuristique

↓

SELECTION de la MEILLEURE ZONE PROMETTEUSE
parmi celles de la liste prometteuse

↓

INTENSIFICATION
à l'intérieur de la meilleure zone prometteuse

2ème Métaheuristique

↓

MEILLEURE SOLUTION RENCONTREE

Une métaheuristique remplace un mécanisme d'une autre

ACO-SA

Le SA remplace le mécanisme de la mise à jour des phéromones

GA-PSO

Le PSO remplace le mécanisme du mutation

5. HYBRIDATION | NIVEAU « COOPÉRATIF »

Pas de relation entre les mécanismes de chaque métaheuristique

Mode séquentiel avec coopération

À t=t1 — Métaheuristique 1

↓

À t=t1+nt — Métaheuristique 2

↓

À t=t1+2nt — Métaheuristique 1

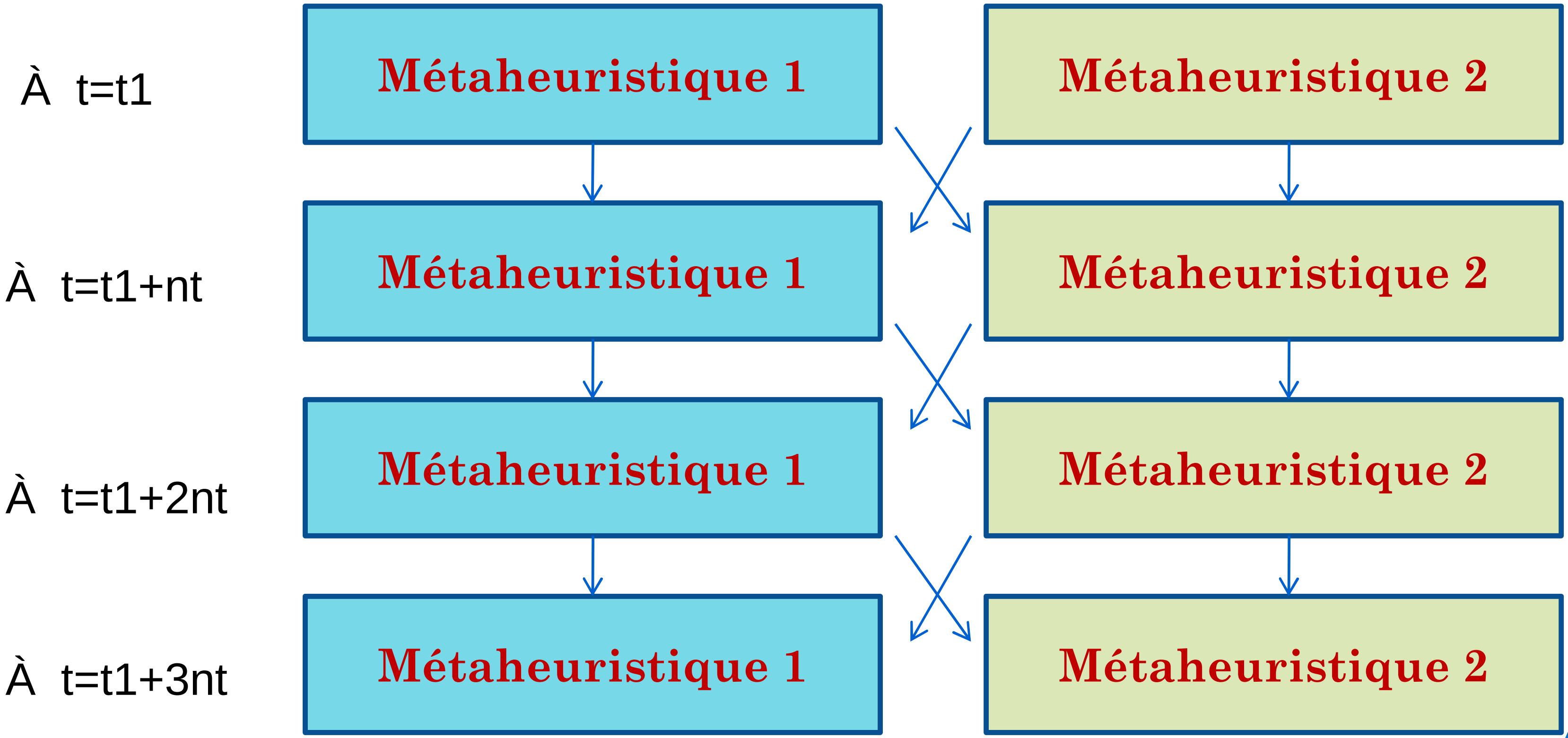
↓

À t=t1+3nt — Métaheuristique 2

↓

Pas de relation entre les mécanismes de chaque métaheuristique

Mode interactif avec coopération



Hybridation de plus de deux métaheuristiques:

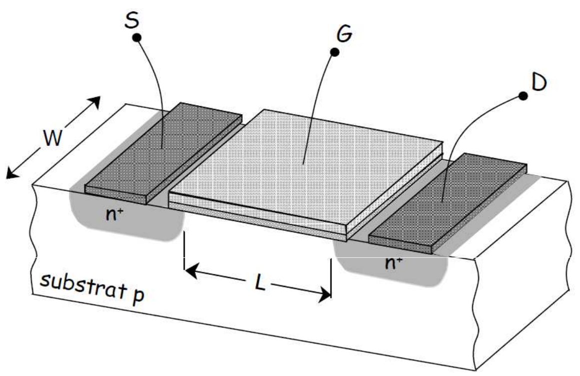
Exemple: Globale globale locale

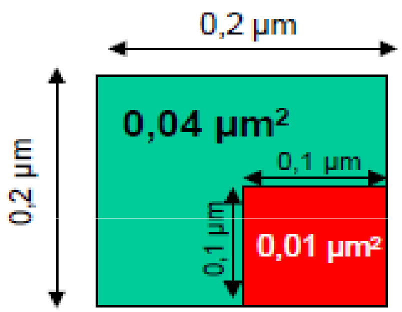
Hybridation avec des algorithme exactes:

- **Nécessité d'une bonne connaissance de l'espace de recherche**
- **Réduction conséquente de l'espace de recherche avant l'application de l'algorithme exacte**

SOMMAIRE

Avancées en matière de la technologie d'intégration la réduction continue des dimensions des composants (essentiellement les transistors)





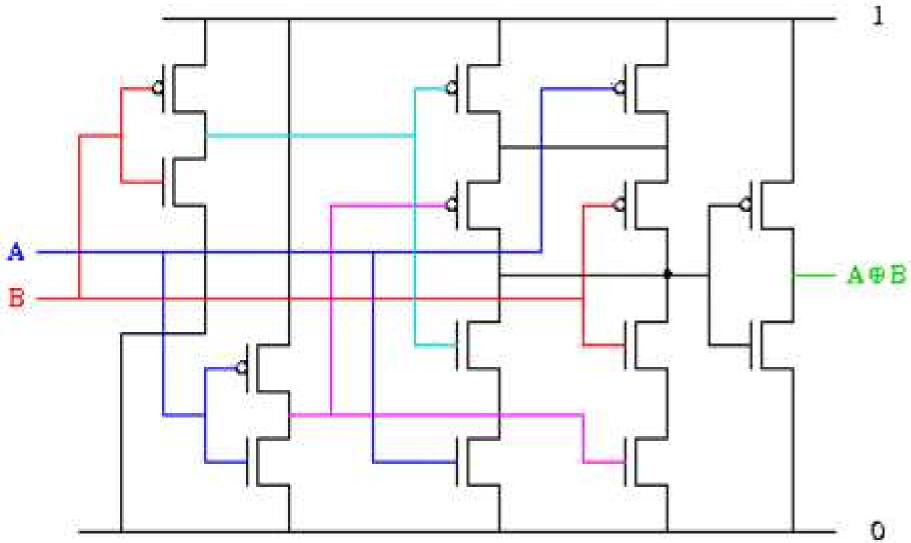
Source: Intel

Problème lors de la conception

6. Application | Problématique

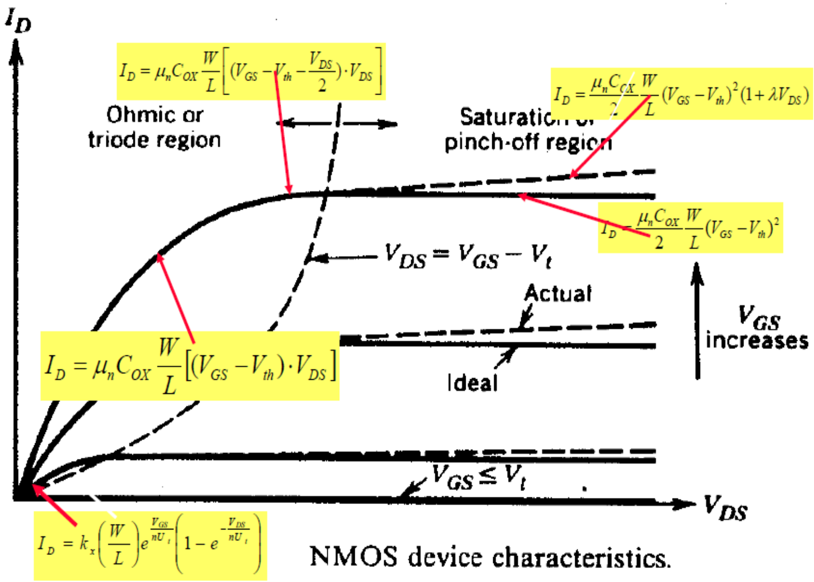
Les circuits numériques

Conception maitrisée



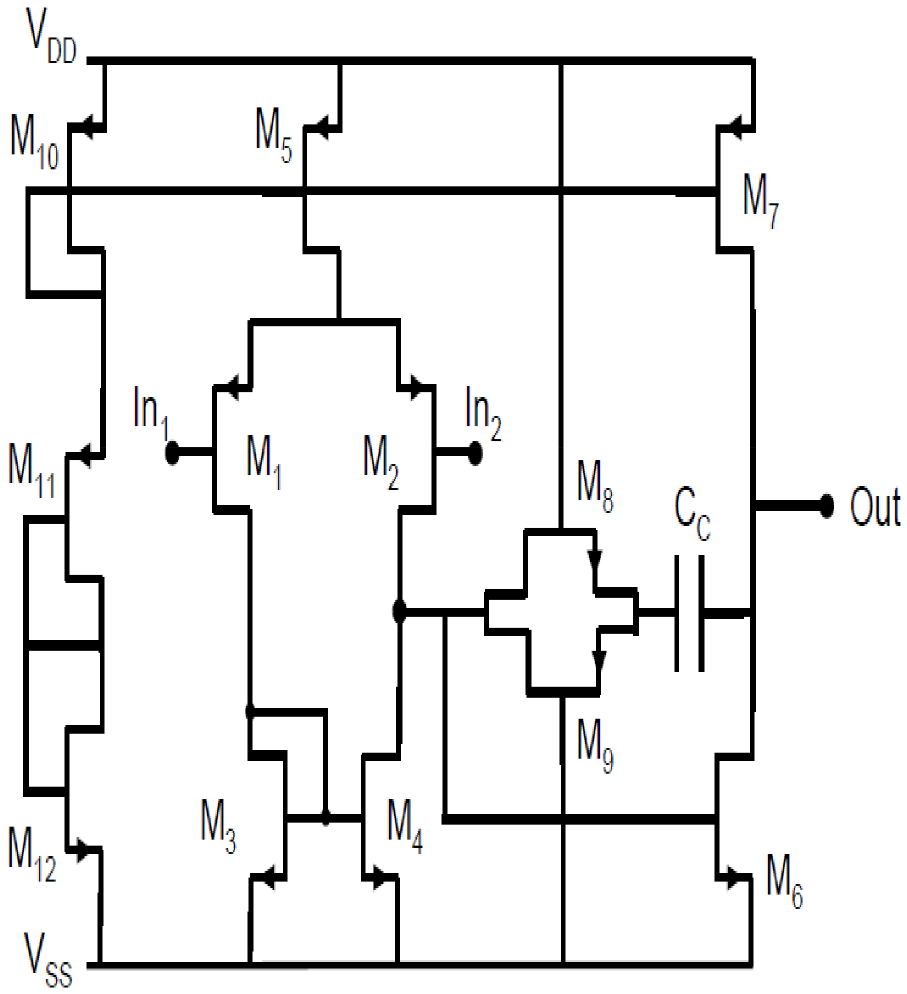
Les circuits analogiques

Conception très difficile



NMOS device characteristics.

Amplificateur opérationnel classique



12 transistors et Cc

25 variables

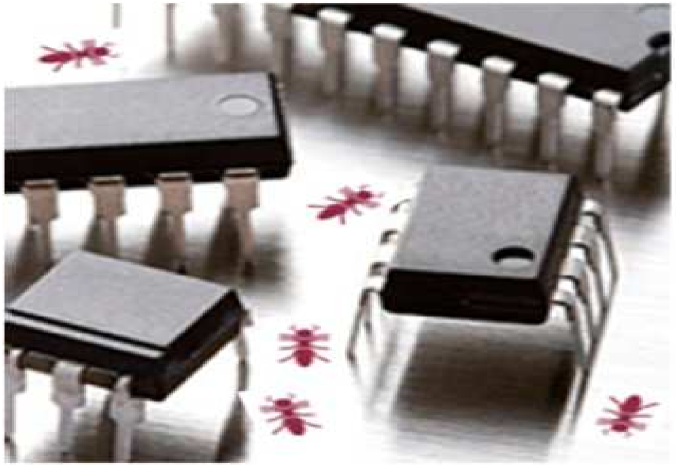
V : $[V_1 , .. , V_{100}]$

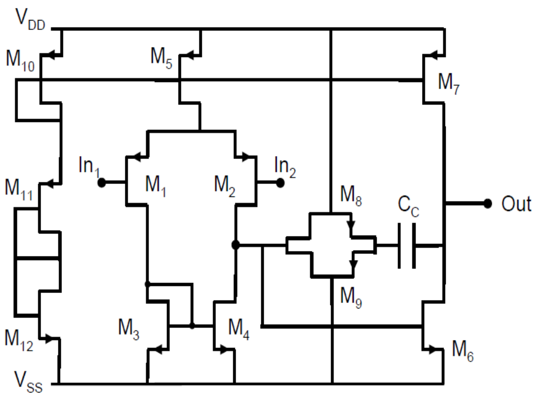
10^{50} Possibilités à vérifier

Très difficile !!!!!!

L'automatisation de la conception des circuits intégrés analogiques ?
Essor avec l'arrivée des métaheuristiques

Appliquer les métaheuristiques au dimensionnement optimal des circuits analogiques





- Fonctions objectifs
- Variables
- Espace de recherche
- Contraintes
- Technologie

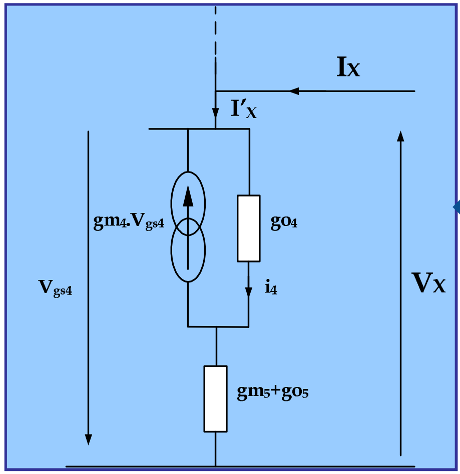
Multi-Objectif

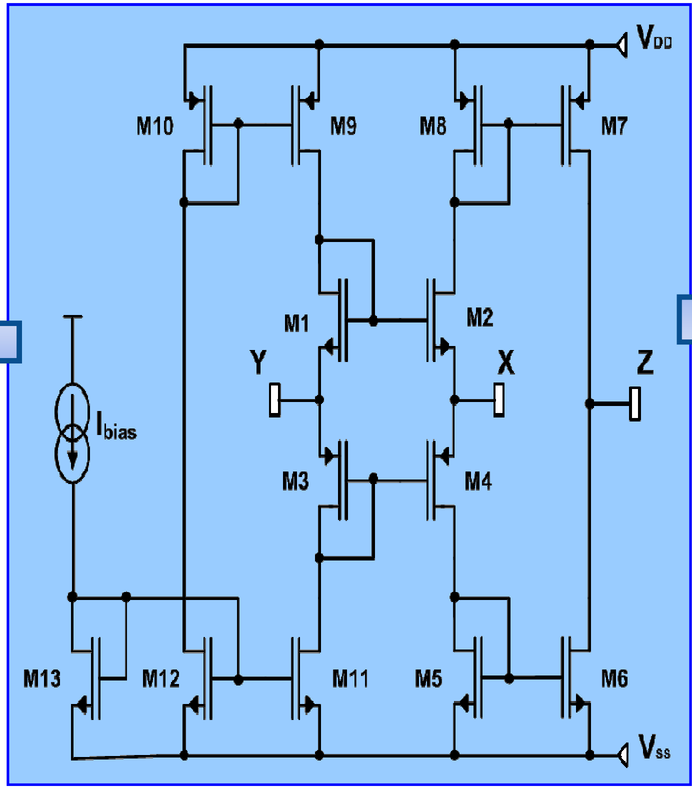
Mono-Objectif

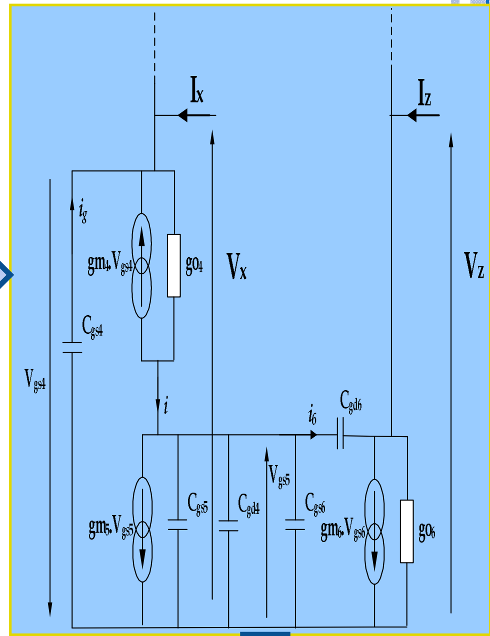
Résultats

Simulations et validation

6. Application | Convoyeur de courant







$$R_X = \frac{gm_N gm_P + (gm_N + gm_P)(go_N + go_P) + (go_N + go_P)^2}{\begin{pmatrix} 3go_N go_P(gm_N + gm_P) + 3gm_N gm_P(go_N + go_P) + 2go_N go_P(go_N + go_P) + \\ 2go_N^2 gm_P + 2go_P^2 gm_N + gm_N^2(go_P + gm_P) + gm_P^2(go_P + gm_N) \end{pmatrix}}$$

$$f_{ci} = \frac{1}{2\pi}\sqrt{\frac{gm_5(gm_4 + go_4)}{Cgs_4(Cgs_5 + Cgs_6 + Cgd_4)}}$$

Mono-objectif

f_{Ci} ↑ R_X ↓

	L_N (µm)	W_N (µm)	L_P (µm)	W_P (µm)	Optimal (ACO)	Simulation PSPICE
$R_{X_{min}}$	0.58	18.57	0.35	30.00	401 Ω	394 Ω
$f_{ci_{max}}$	0.56	05.39	0.35	09.49	1.791 GHz	1.817 GHz

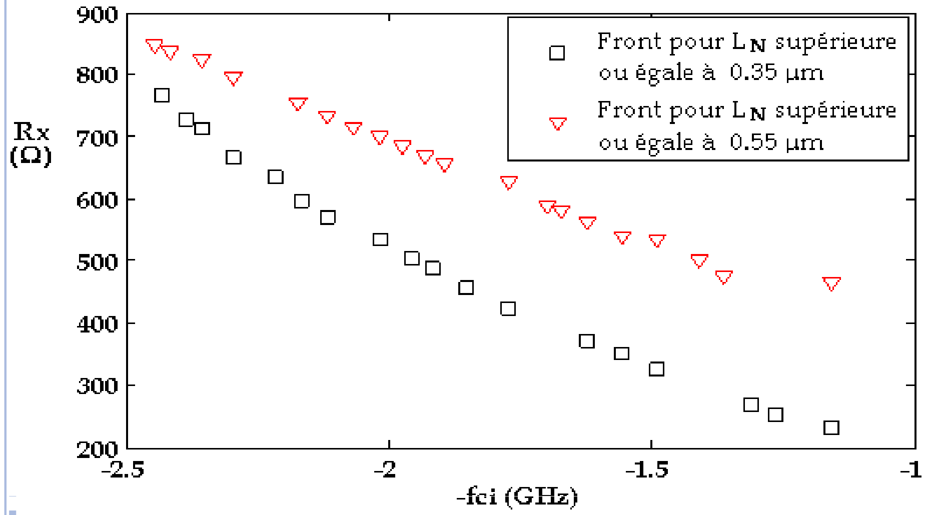
	L_N (µm)	W_N (µm)	L_P (µm)	W_P (µm)	$R_{X_{min}}$ (Ω)	L_N (µm)	W_N (µm)	L_P (µm)	W_P (µm)	$f_{ci_{max}}$ (GHz)
PSO	0.59	18.74	0.35	30.00	**464**	0.54	05.80	0.35	10.05	**1.751**
ACO	0.58	18.57	0.35	30.00	**401**	0.56	05.39	0.35	09.49	**1.791**
GA	0.50	13.86	0.35	27.32	**529**	0.58	5.96	0.36	10.56	**1.576**
SA	0.55	17.49	0.35	30.00	**473**	0.55	8.16	0.35	14.13	**1.422**

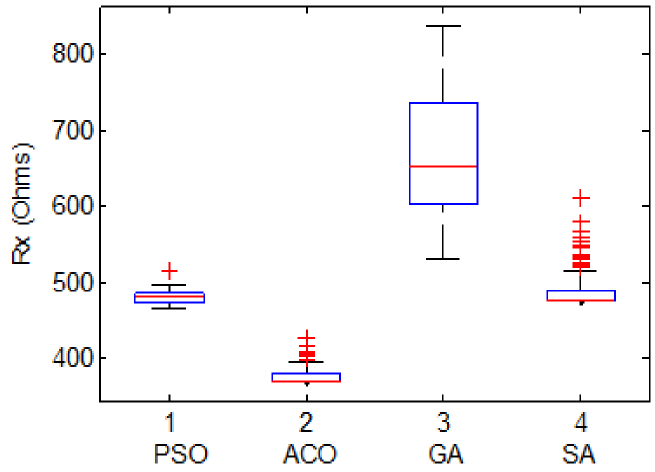
Pondération

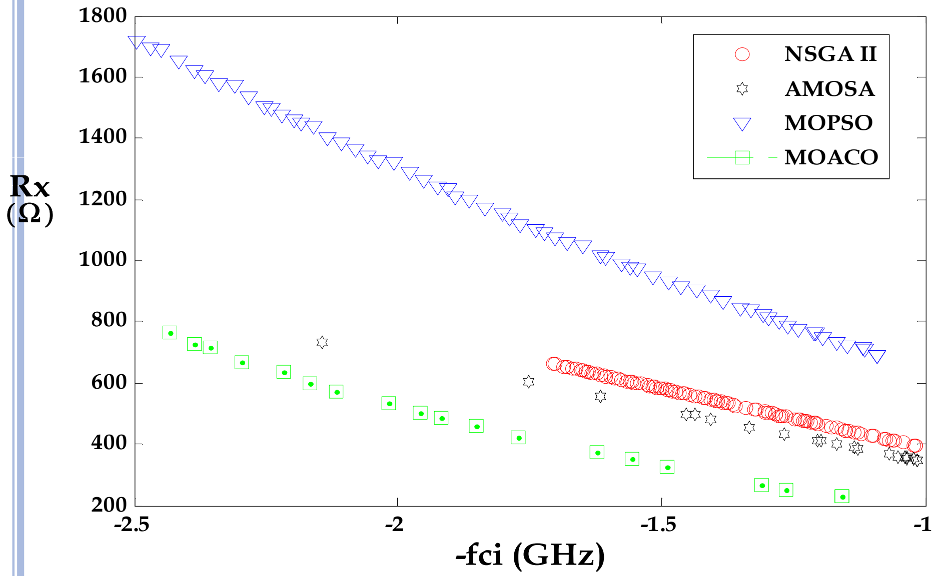
$$FO = \delta_1 R_X + \delta_2 f_{Ci}$$

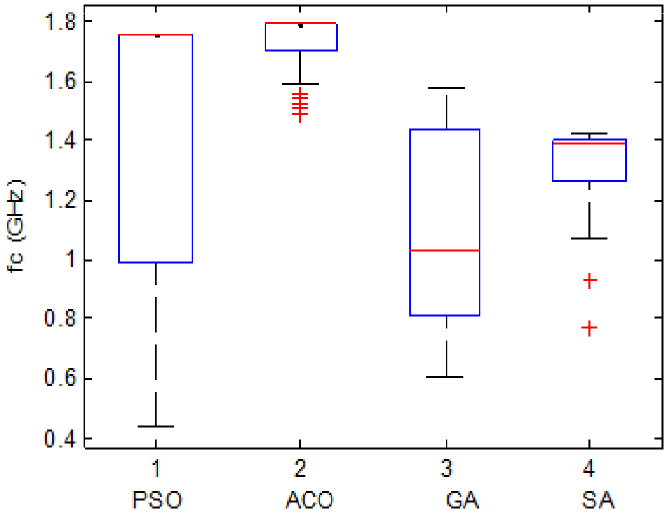
$L_N = L_P$ (µm)	W_N (µm)	W_P (µm)	R_X (Ω)		f_{ci} (GHz)	
			Optimal (ACO)	Simulation PSPICE	Optimal (ACO)	Simulation PSPICE
0.35	23.56	45.12	477	489	1.873	1.864

Multi-objectifs







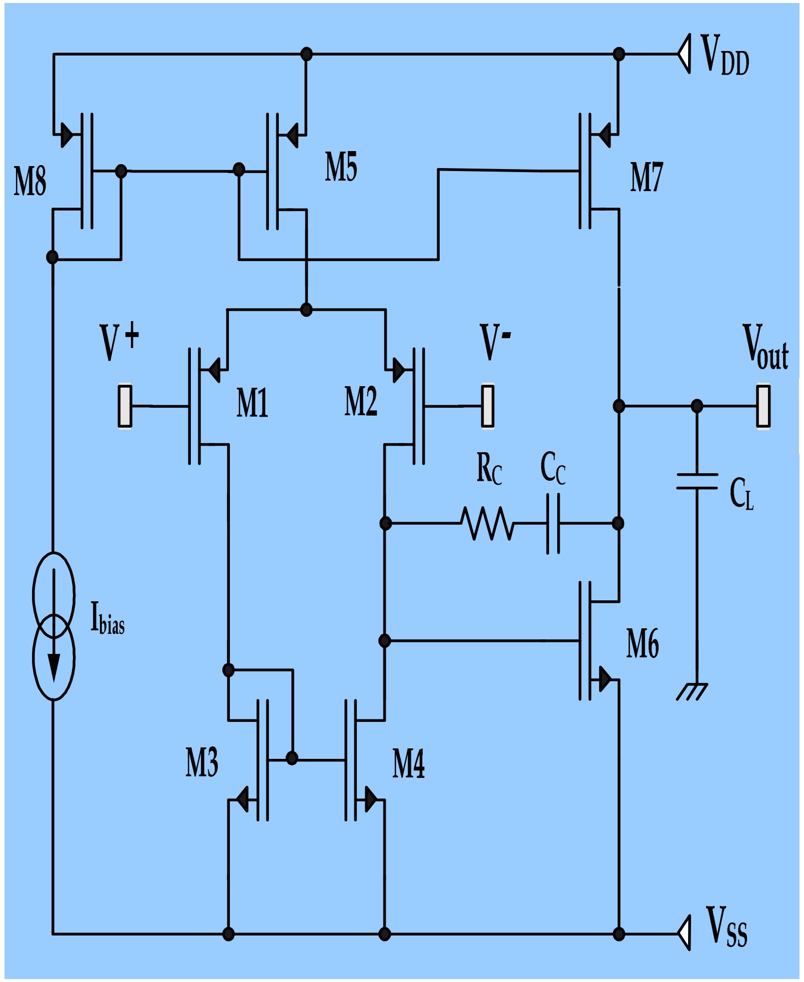


$$Av = \frac{gm_2 gm_6}{(go_2 + go_4)(go_6 + go_7)}$$

$$CMRR = \frac{A_d}{A_c} = \frac{2gm_1 gm_2}{go_7(go_2 + go_4)}$$

$$SR = min\left\{\frac{2I_1}{C_c}, \frac{I_7}{C_L + C_{gd_6} + C_{gs_7}}\right\}$$

$$F_c = \frac{gm_1}{2\pi C_c}$$



$$A \approx \sum_{i=1}^{8} W_i L_i$$

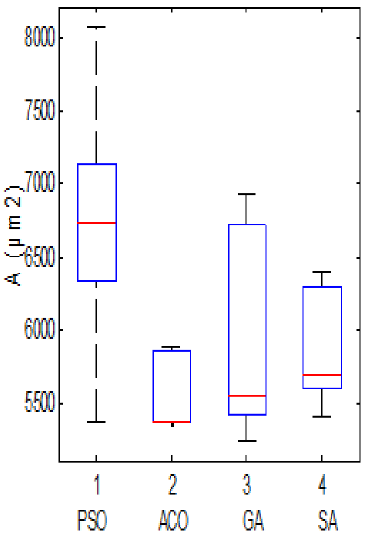
$$P = (V_{DD} - V_{SS})(I_{bias} + I_5 + I_7)$$

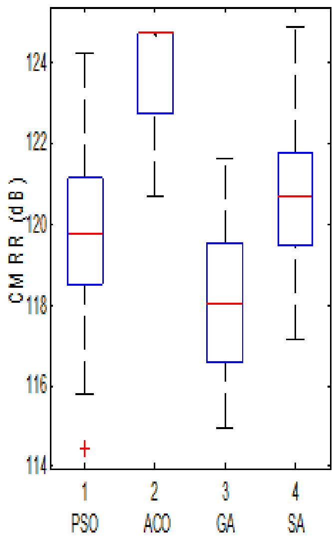
Mono-objectif

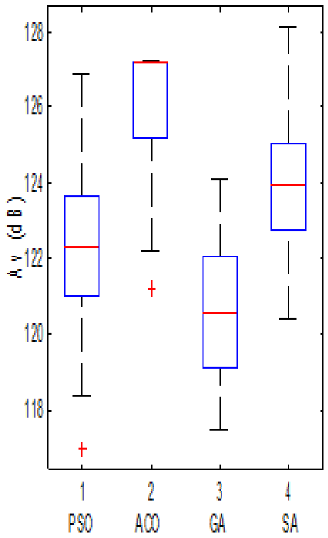
$$FO = \delta_1 A_V + \delta_2 F_C + \delta_3 SR + \delta_4 CMRR + \delta_5 A + \delta_6 P$$

$W_{1,2}$ (µm)	$W_{3,4}$ (µm)	W_5 (µm)	W_6 (µm)	W_7 (µm)	W_8 (µm)	L (µm)	Io (µA)	Cc (pF)	Rc (Ω)
201.00	130.40	51.20	570.40	111.70	5.00	0.80	10.00	09.20	236

	Av (dB)	CMRR (dB)	P (mW)	A (µm²)	Fc (MHz)	SR (V/µs)
Optimal (ACO)	102.10	106.00	1.70	1124	98.14	12.32
Simulation PSPICE	95.60	98.61	2.18	-----	93.54	11.54

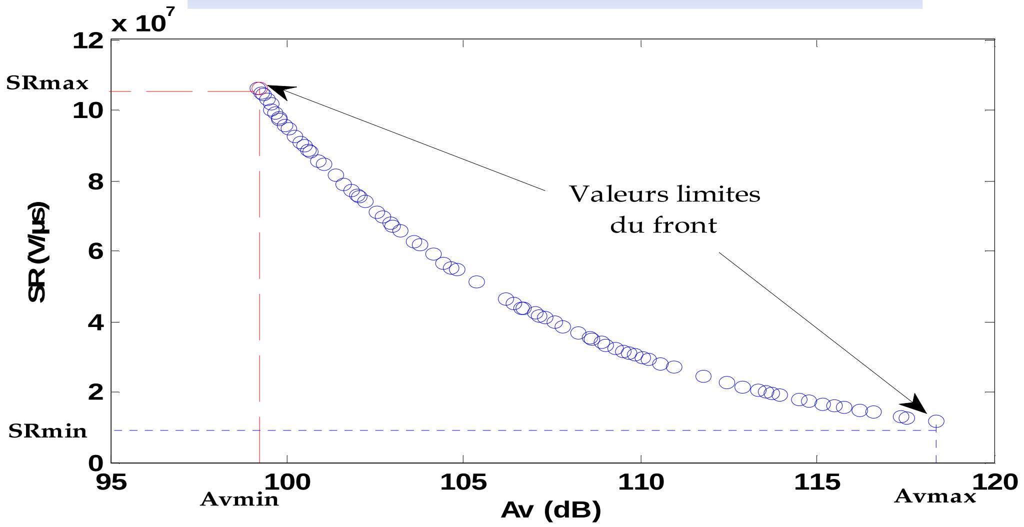


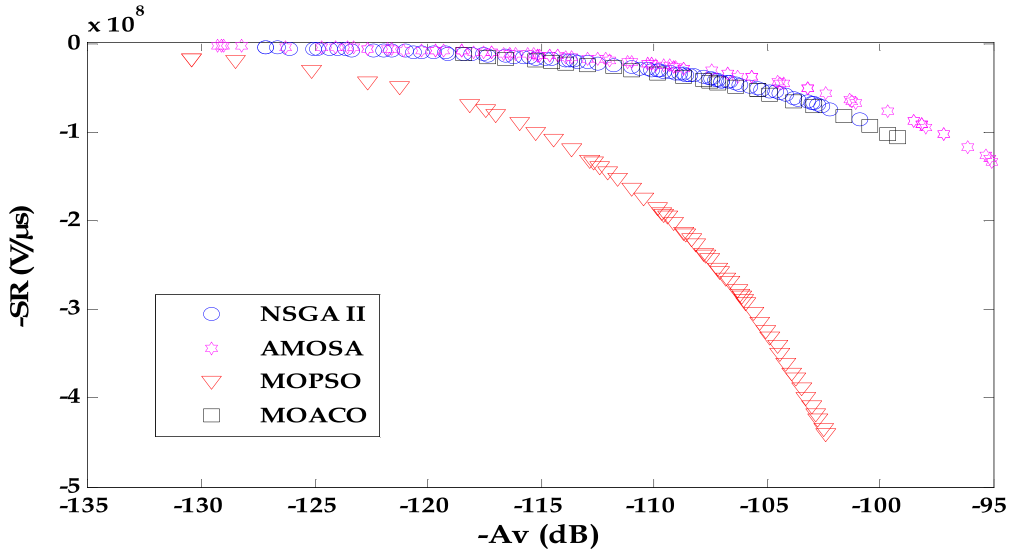




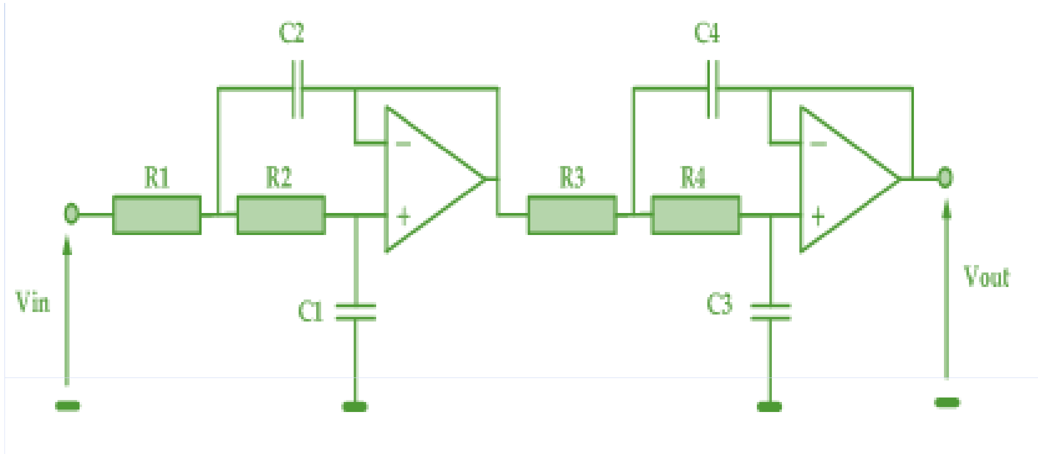
	Av (dB)	CMRR (dB)	A (µm²)	P (mW)
PSO	126.73	124.23	5368	1.3
ACO	127.22	124.72	5354	1.2
GA	124.12	121.62	5242	1.6
SA	127.36	124.86	5408	1.2

Multi-objectifs





Butterworth du 4ème ordre



$$Error_{total} = \left(0.5\,\frac{\left|\frac{1}{\sqrt{R_1R_2C_1C_2}} - \omega_c\right| + \left|\frac{1}{\sqrt{R_3R_4C_3C_4}} - \omega_c\right|}{\omega_c} + 0.5\left(\left|\frac{\sqrt{R_1R_2C_1C_2}}{R_1C_1 + R_2C_1} - \frac{1}{0.7654}\right| + \left|\frac{\sqrt{R_3R_4C_3C_4}}{R_3C_3 + R_4C_3} - \frac{1}{1.8478}\right|\right)\right)$$

	R1 (KΩ)	R2 (KΩ)	C1 (nF)	C2 (nF)	R3 (KΩ)	R4 (KΩ)	C3 (nF)	C4 (nF)	$\Delta\omega$	ΔQ	**Erreur totale**
Valeurs linéaires	4.6768	4.9495	7.8788	53.9394	1.00	2.3333	55.00	78.00	0.0087	0.0058	**0.0072**
E12	4.70	4.70	8.20	56.00	1.00	2.20	56.00	82.00	0.0122	0.0198	**0.0160**
E24	4.70	5.10	8.20	56.00	1.00	2.40	56.00	75.00	0.0508	0.0148	**0.0328**
E48	4.64	4.87	7.87	53.60	1.00	2.37	56.20	78.70	0.0475	0.0026	**0.0251**
E96	4.64	4.99	7.87	53.60	1.00	2.32	54.90	78.70	0.0131	0.0106	**0.0118**
E192	4.70	4.99	7.87	54.20	1.00	2.34	54.90	77.70	0.0011	0.0087	**0.0049**

	R1 (KΩ)	R2 (KΩ)	C1 (nF)	C2 (nF)	R3 (KΩ)	R4 (KΩ)	C3 (nF)	C4 (nF)	$\Delta\omega$	ΔQ	**Erreur totale**
GA	6.80	6.80	5.60	39.00	39.00	1.00	4.70	56.00	0.0179	0.0153	**0.0166**
PSO	4.58	4.70	8.20	56.00	1.10	1.00	87.60	102.20	0.0135	0.0018	**0.0076**
ACO	4.70	4.99	7.87	54.20	1.00	2.34	54.90	77.70	0.0011	0.0087	**0.0049**

SOMMAIRE

Conclusion

- Les métaheuristiques: Domaine de recherche riche et en constante évolution (revues, conférences spécialisées et nombre d'articles)
- Etant donné un problème d'optimisation, quelle méthode choisir ?
  - pour produire une solution optimale (ou acceptable)
  - avec un temps de calcul raisonnable ?
- Pas de recette miracle !
- On dispose de plusieurs méthodes, choisir la mieux adaptée au problème d'optimisation : (espace de recherche, fonction objectif ...)

- Bonne fomalisation/modélisation du problème
- Adaptation "intelligente" d'une métaheuristique : intégration des connaissances spécifiques + structures de données efficaces

- Tendance actuelle : Hybridation / parallélisme

RÉFÉRENCES

[1] DORIGO, Marco, GAMBARDELLA, Ant Colony System : A Cooperative Learning Approch to the Traveling Salesman Problem. IEEE Transactions on Evolutionary Computation, Vol : 1 No : 1. p.53-66, 1997

[2] B. Bullnheimer, R. F. Hartl and C. Strauss, A New Rank Based Version of the Ant System: A Computational Study. Central European Journal for Operations Research and Economics, Vol:7. p.25-38, 1999

[3] Christian, Blum, Ant colony optimization: Introduction and recent trends. Physics of Life Reviews, Vol: 2. p.353-373, 2005

[4] Tony, White, Simon, Kaegi, Terri, Oda, Revisiting Elitism in Ant Colony Optimization. Genetic and Evolutionnary Computation Conference, LNCS 2723, p.122-133, 2003

[5] Kennedy, J. (1997), "The Particle Swarm: Social Adaptation of Knowledge", Proceedings of the 1997 International Conference on Evolutionary Computation, pp. 303-308, IEEE Press.]

[6] Carlisle, A. and Dozier, G. (2001). "An Off-The-Shelf PSO", Proceedings of the 2001 Workshop on Particle Swarm Optimization, pp. 1-6, Indianapolis, IN.

[7] A Bee Algorithm for Multi-Agents System-Lemmens ,Steven . Karl Tuyls, Ann Nowe - 2007

[8] Kirkpatrick, S., Gelatt. C. D., and Vecchi, M. P. "Optimization by simulated annealing", Science, 13 May 1983 220, 4598, 671–680, 1983.

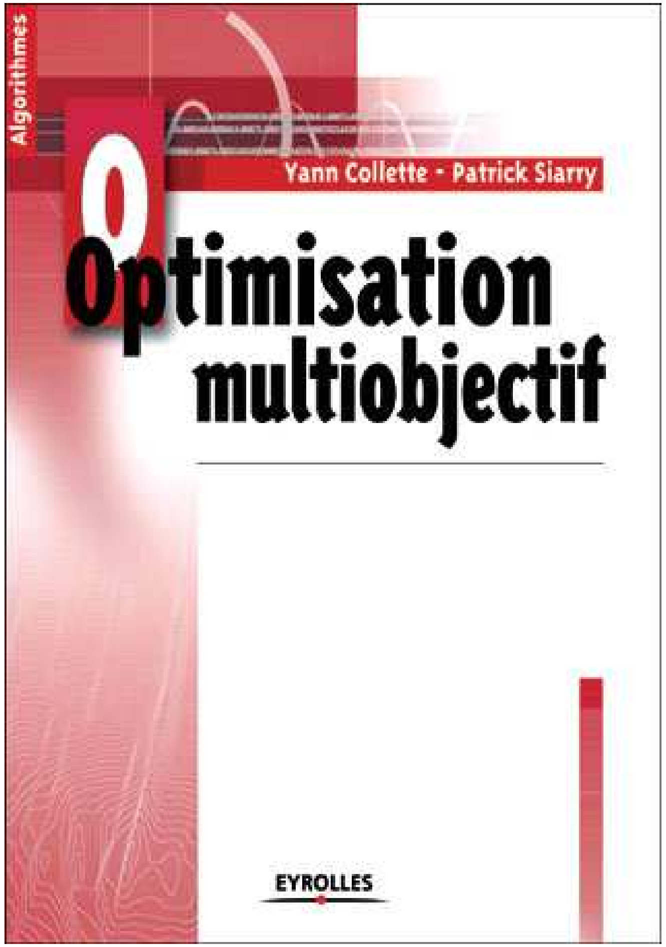
[18] D. Karaboga, B. Basturk. "On the performance of artificial bee colony (ABC) algorithm," Elsevier, Applied Soft Computing, vol. 8, No. 1, January 2008, Pages 687–697.

http://animatlab.lip6.fr/papers/Mouret_LMag_05_ga.pdf

http://magnin.plil.net/spip.php?rubrique8

http://fr.wikipedia.org/wiki/Algorithme_g%C3%A9n%C3%A9tique

LIVRES RECOMMANDÉS



Johann Dréo,
Alain Pétrowski,
Patrick Siarry,
Eric Taillard

Yann Collette,
Patrick Siarry

Editeur : Eyrolles

MERCI DE VOTRE ATTENTION