L'IA pour les ressources en eau et l'hydrologie :

Gestion optimale des réservoirs et prévision des apports par algorithmes génétiques et réseaux de neurones face au changement climatique

D Nagesh Kumar

Professeur, Département de génie civil

Indian Institute of Science, Bangalore

URL: http://www.civil.iisc.ac.in/~nagesh

Remerciements : ***Pr M Janga Reddy***, IIT Bombay

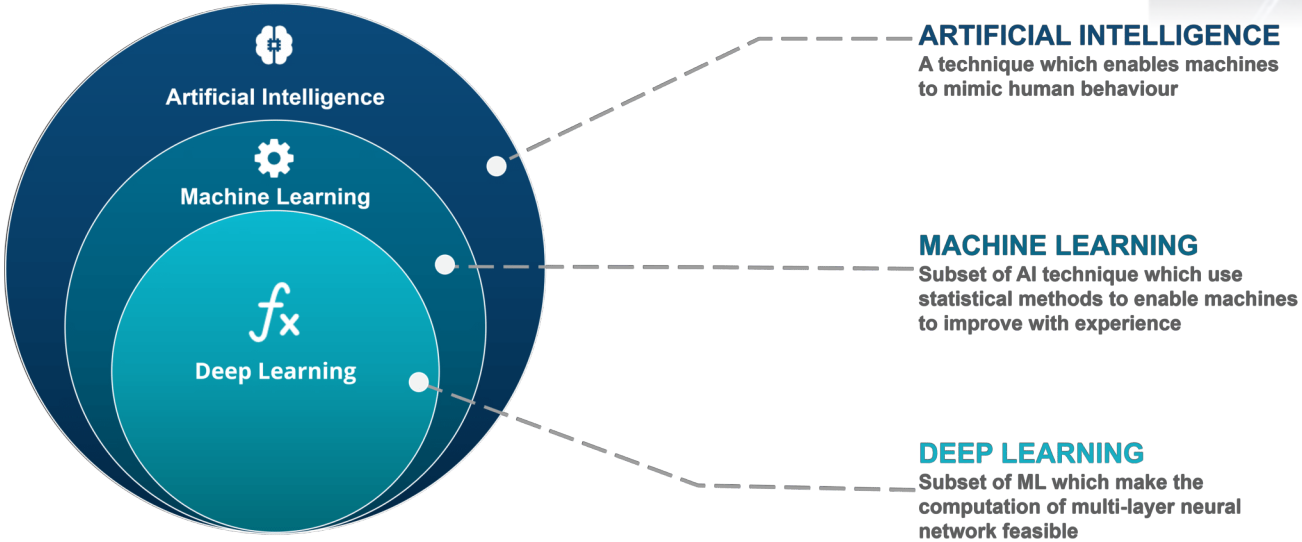
Plan

- Méthodes d'optimisation
- Algorithmes génétiques (AG)
  - Introduction
  - Les AG pour l'optimisation
  - Études de cas de gestion de réservoirs mono- et multi-objectif
- Réseaux de neurones artificiels pour la prévision des apports face au changement climatique
  - RNA
  - RN-PSO
  - ANFIS
- Conclusions et perspectives

Apprentissage profond



Source : Université Harvard

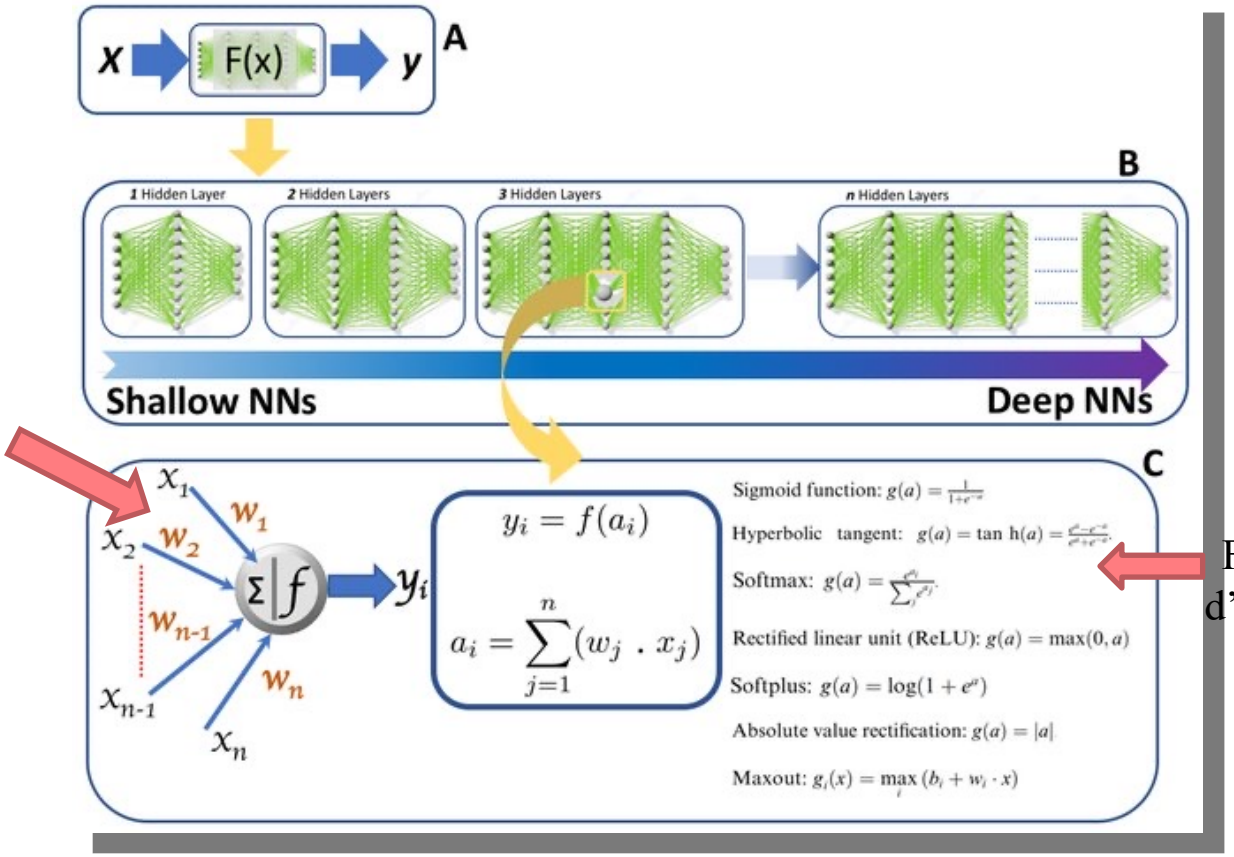


Source : Edureka

- **L'apprentissage profond** est un algorithme informatique qui imite le réseau de neurones du cerveau.
- C'est une branche de l'apprentissage automatique fondée sur les réseaux de neurones artificiels.
- On parle d'apprentissage profond parce qu'il utilise des réseaux de neurones profonds.
- Cet apprentissage peut être supervisé, semi-supervisé ou non supervisé.

Réseaux de neurones artificiels

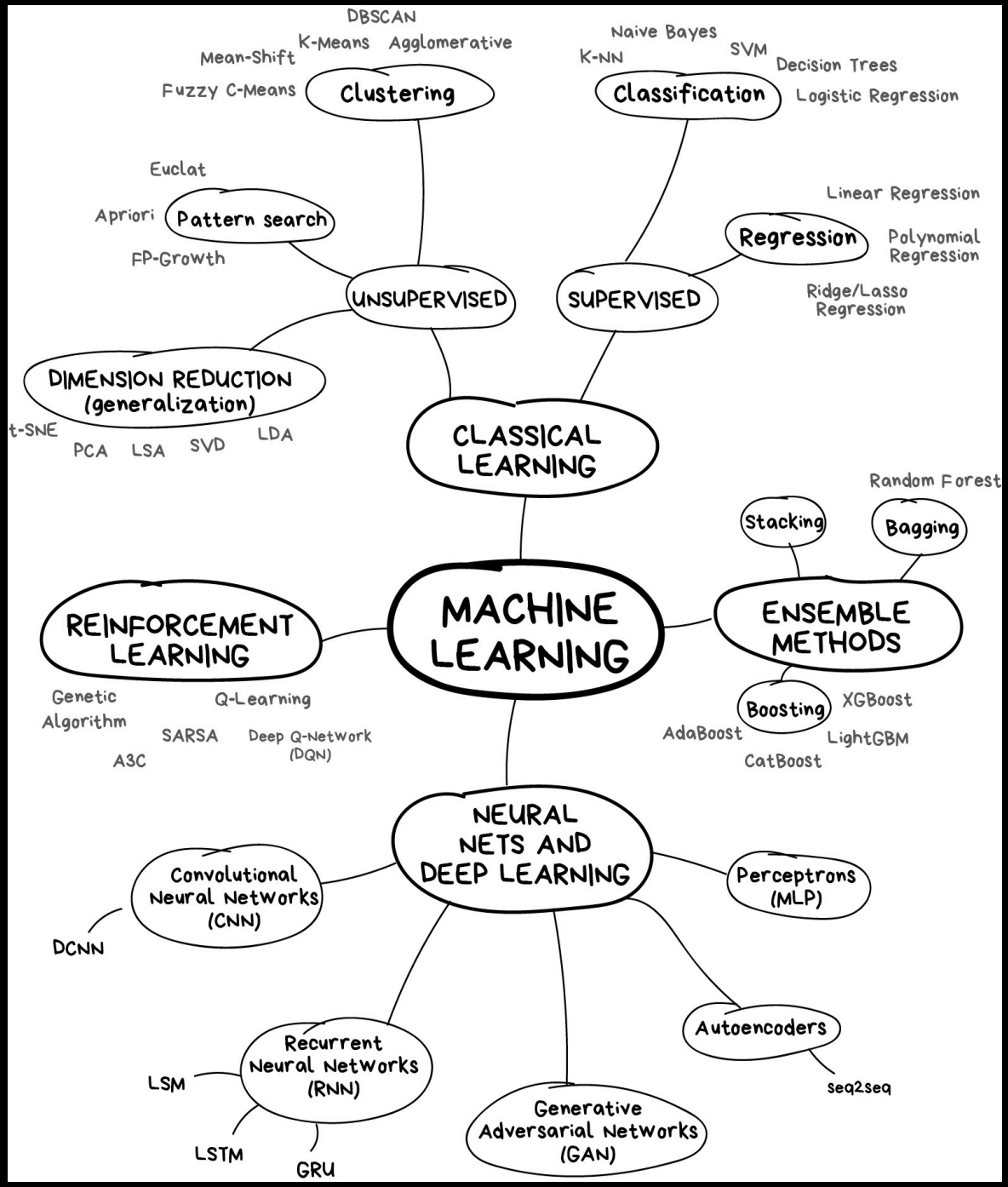
Un **neurone artificiel** est une fonction mathématique fondée sur un modèle des neurones biologiques : chaque neurone reçoit des entrées, les pondère séparément, les additionne et fait passer cette somme dans une fonction non linéaire pour produire une sortie.



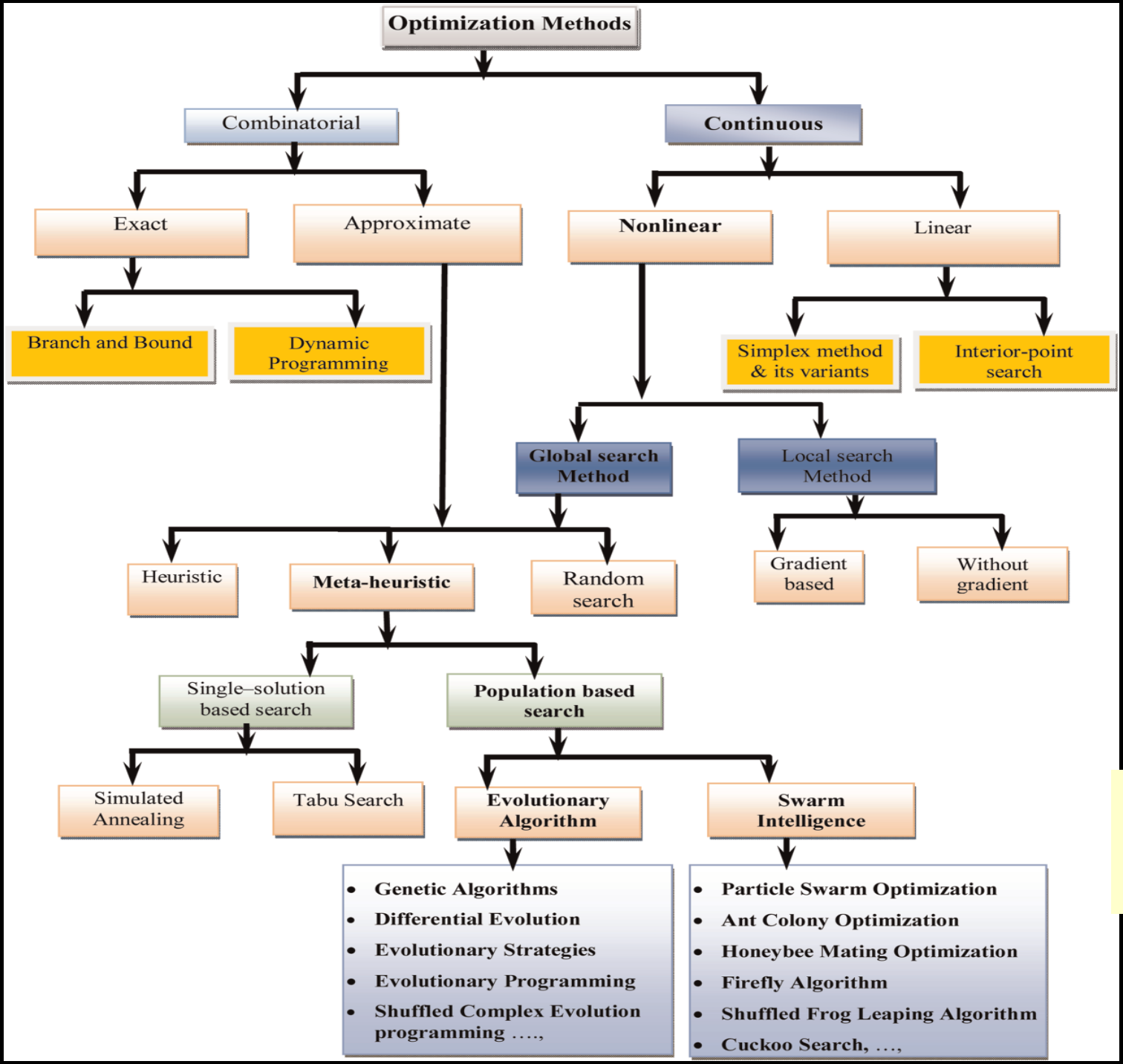
Fonctions d'activation

Zemouri et al., 2019

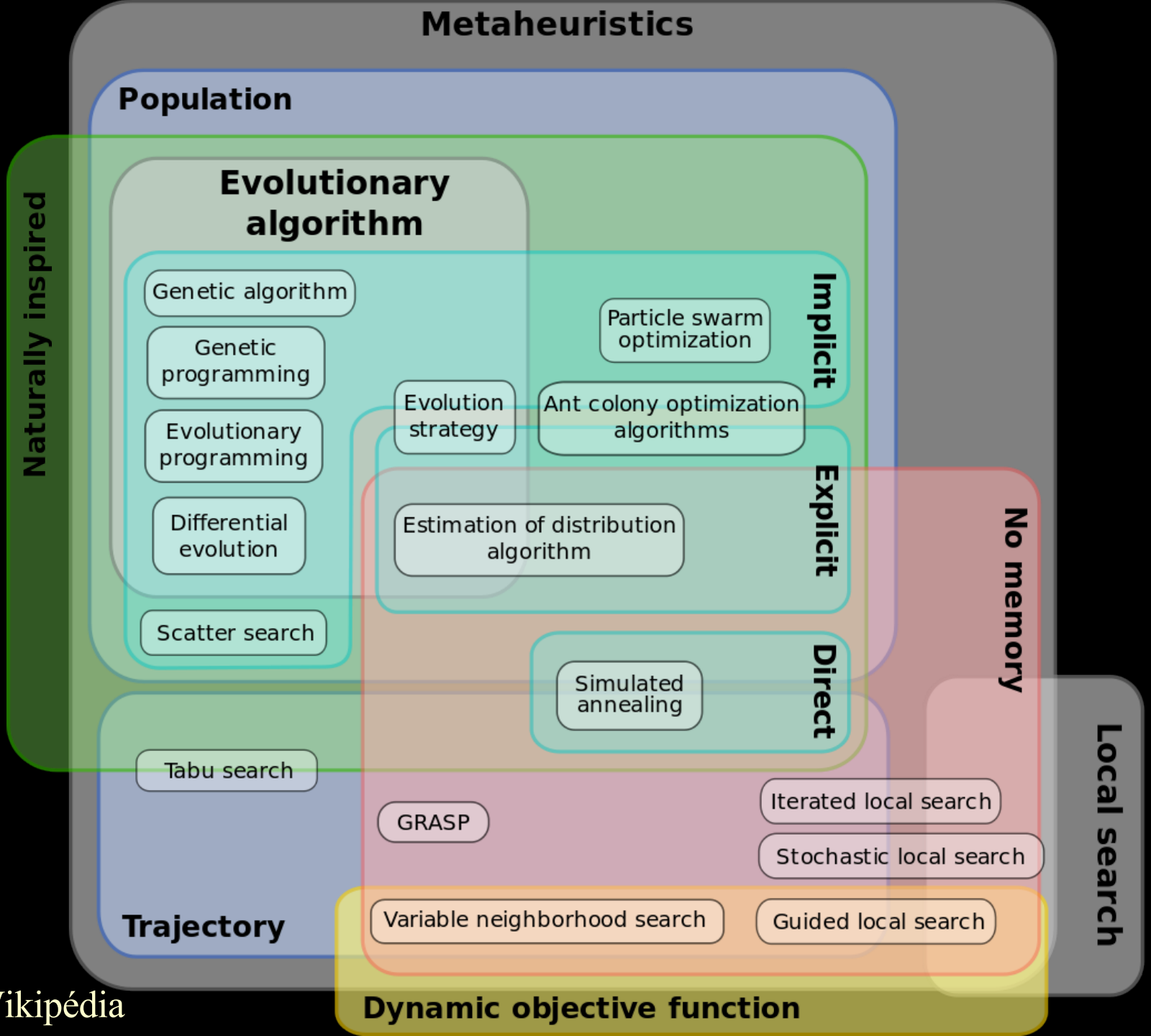
Carte détaillée de l'apprentissage automatique



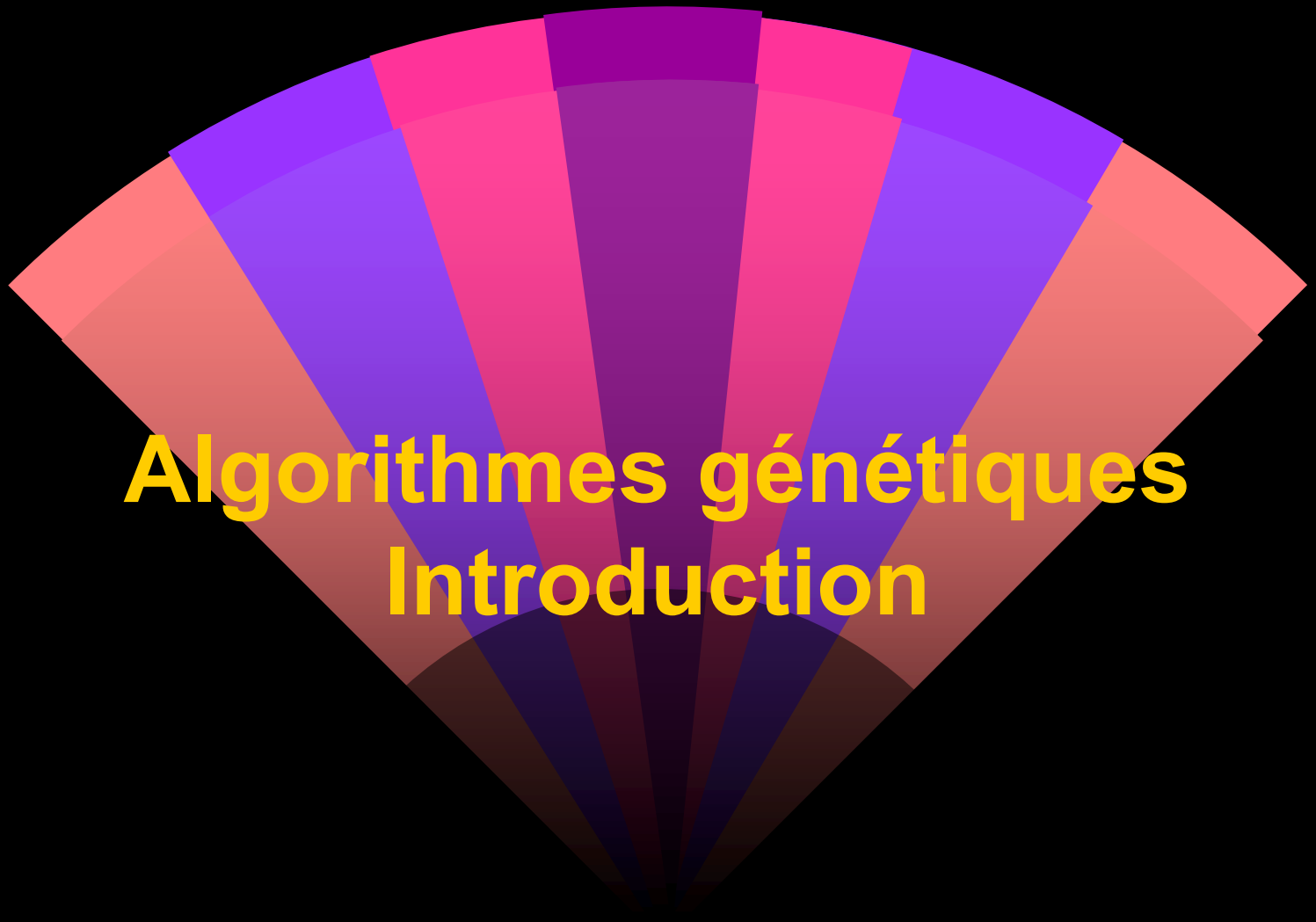
Source : https://vas3k.com/blog/machine_learning/



Janga Reddy et Nagesh Kumar, 2020



Source : Wikipédia



Algorithmes génétiques
Introduction

Introduction



Introduction

Inspirés de l'évolution naturelle

- Les êtres vivants ont évolué vers des organismes plus performants
  - La descendance présente certains traits de chaque parent
  - Les traits héréditaires sont déterminés par les gènes
  - Les instructions génétiques sont contenues dans les chromosomes
  - Les chromosomes sont des brins d'ADN
  - L'ADN est composé de paires de bases qui, combinées de façon significative, codent les traits héréditaires

Mécanismes du changement évolutif

- **Croisement**

  Échange aléatoire de chromosomes entre parents lors de la reproduction : la descendance hérite de traits de chaque parent

Le croisement exige une diversité génétique entre parents pour garantir une descendance suffisamment variée

Mécanismes du changement évolutif

- **Mutation**

  Erreurs rares lors de la copie des chromosomes, qui entraînent

  - des changements absurdes/mortels, produisant des organismes non viables
  - des changements bénéfiques, produisant des organismes « plus forts »
  - des changements ni nuisibles/mortels ni bénéfiques, produisant des organismes non améliorés

Mécanismes du changement évolutif

- **Sélection naturelle**

  Dans un environnement concurrentiel, les plus aptes survivent, ce qui donne de meilleurs organismes

  - Les individus aux meilleurs traits de survie vivent généralement plus longtemps
  - Ils ont ainsi plus de chances de se reproduire et de transmettre leurs traits avantageux
  - Au fil des générations, l'espèce s'améliore, car les meilleurs traits finissent par dominer les plus faibles

L'approche génétique

- Maintenir une population d'individus qui sont des solutions complètes ou partielles
- Explorer l'espace des solutions en faisant interagir et rivaliser ces individus
  - L'interaction produit de nouveaux individus
  - La compétition élimine les individus faibles
- Après plusieurs générations, on devrait obtenir un individu fort (c.-à-d. une solution)

Terminologie

- **Population**

  Ensemble des solutions possibles d'un problème

- **Élitisme**

  Permettre aux meilleurs membres de passer à la génération suivante

- **Mutation**

  Modifier un seul membre de la population

- **Croisement**

  Combiner deux membres de la population

Terminologie – suite

- **Sélection**

  Déterminer quels membres de la population survivront à la génération suivante

- **Fonction d'adaptation (fitness)**

  Évaluer la qualité d'un membre de la population

- **Schémas de codage**

  Représentation binaire d'un membre de la population

Algorithmes génétiques

- Un outil d'optimisation non linéaire fondé sur des méthodes d'optimisation combinatoire

- Différences avec les méthodes classiques
  - Ils travaillent sur un codage des jeux de paramètres, pas sur les paramètres eux-mêmes
  - Ils n'utilisent que la fonction objectif, sans dépendre de la continuité, des dérivées ou d'autres informations

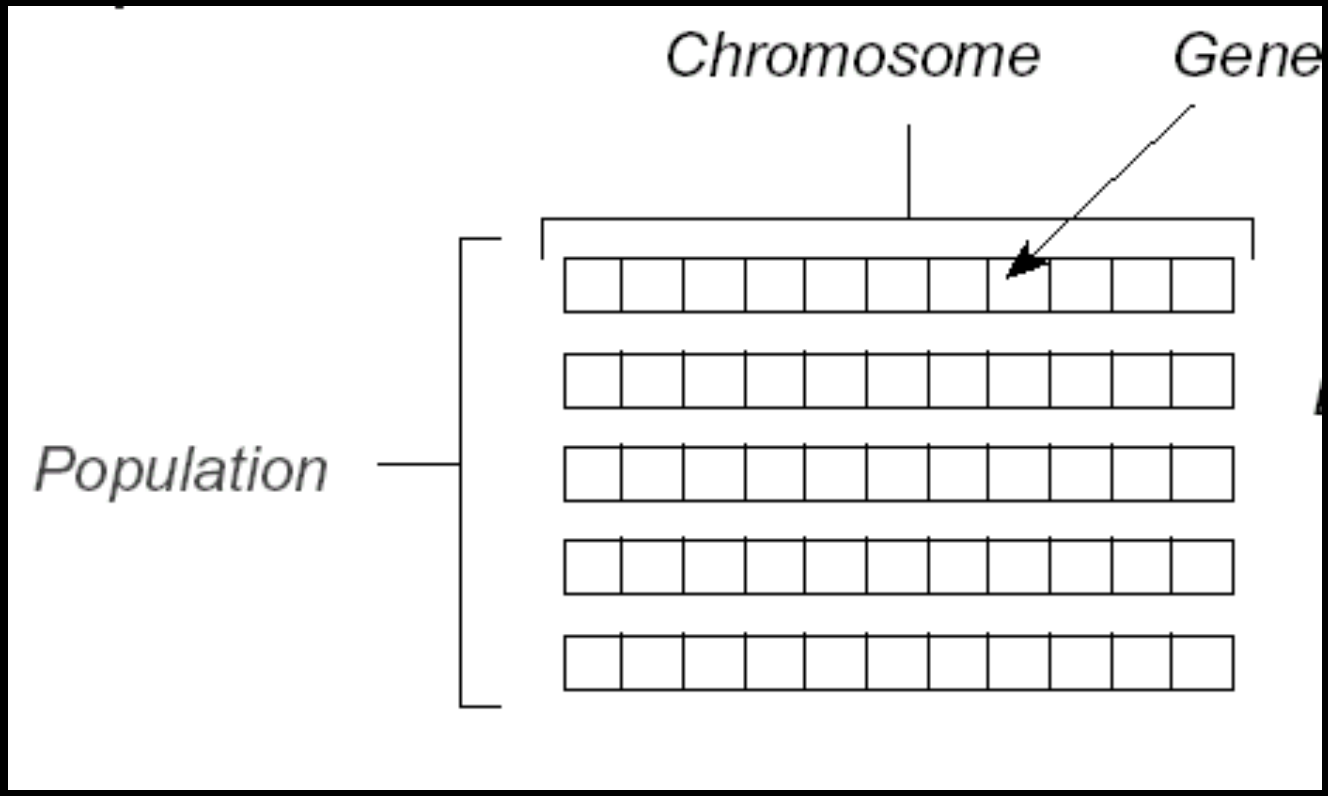
Principe de fonctionnement

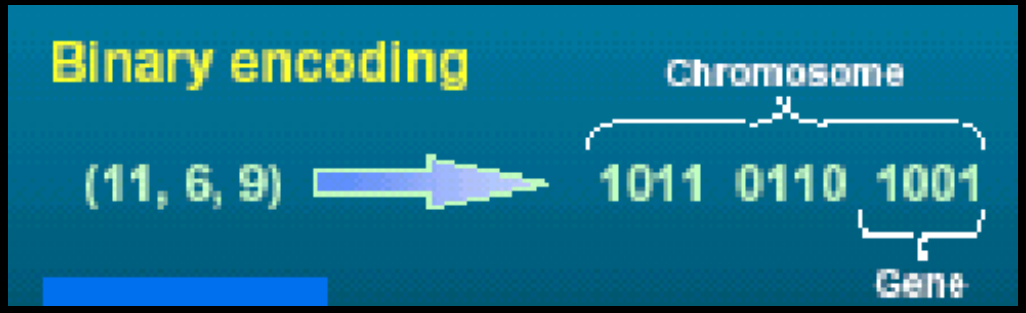
- **Codage**
- **Évaluation de l'adaptation**
- **Opérations génétiques**
  - ***Reproduction***
  - ***Croisement***
  - ***Mutation***
- **Critère d'arrêt**

Codage

- Les variables de décision sont codées en une chaîne binaire appelée chromosome
- La longueur du chromosome dépend de la précision voulue
- Les variables de décision n'ont pas toutes besoin de sous-chaînes de même longueur

Codage – suite





Initialisation : ensemencer la population

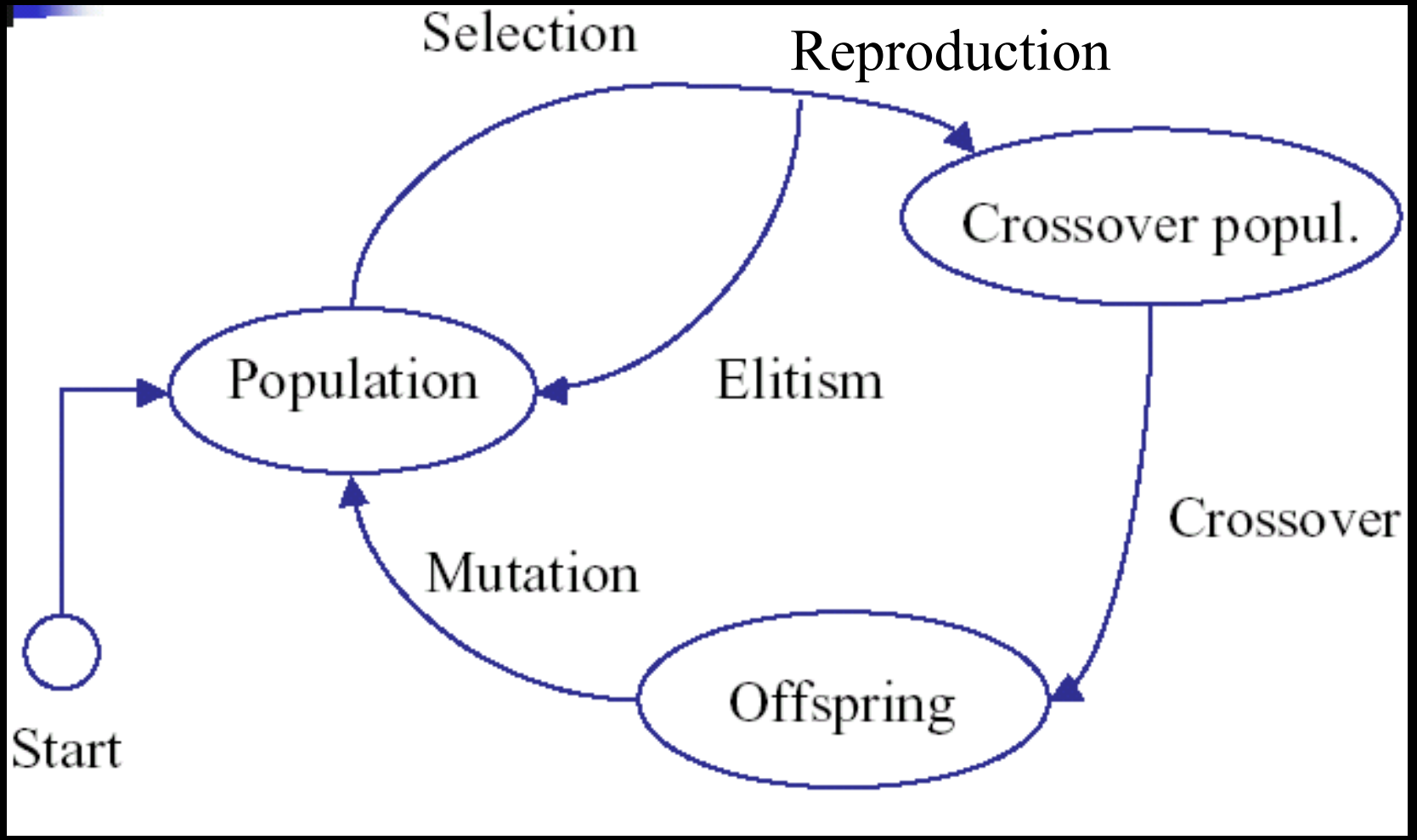
- L'initialisation fixe la population de départ dont seront issues les générations futures

- Points d'attention :
  - Taille de la population initiale
  - Déterminée expérimentalement selon le problème
  - Diversité de la population initiale (diversité génétique)
  - Un problème courant dû au manque de diversité est la convergence prématurée vers une solution non optimale

Évaluation de l'adaptation

- L'évaluation classe les individus selon une mesure d'adaptation qui reflète la qualité de chaque solution.
- Dérivée de la fonction objectif
- La valeur de la fonction d'adaptation d'une chaîne est appelée adaptation de la chaîne
- Toutes les variables de décision sont codées dans une seule chaîne (chromosome)
- Par exemple, pour un individu i :
  - Classification : $(correct(i))^2$
  - TSP : $distance(i)$
  - SAT : $nb\ de\ termes\ satisfaits(i)$
  - animation de marche : *note subjective*
- Les fonctions d'adaptation ont souvent des minima locaux

Cycle d'un AG


Selection
Reproduction
Crossover popul.
Population
Elitism
Crossover
Mutation
Offspring
Start

Techniques de sélection

- Exemple de sélection proportionnelle à l'adaptation
  - **Soit la population**

Individual	Fitness	Prob.
A	5	10%
B	20	40%
C	11	22%
D	8	16%
E	6	12%

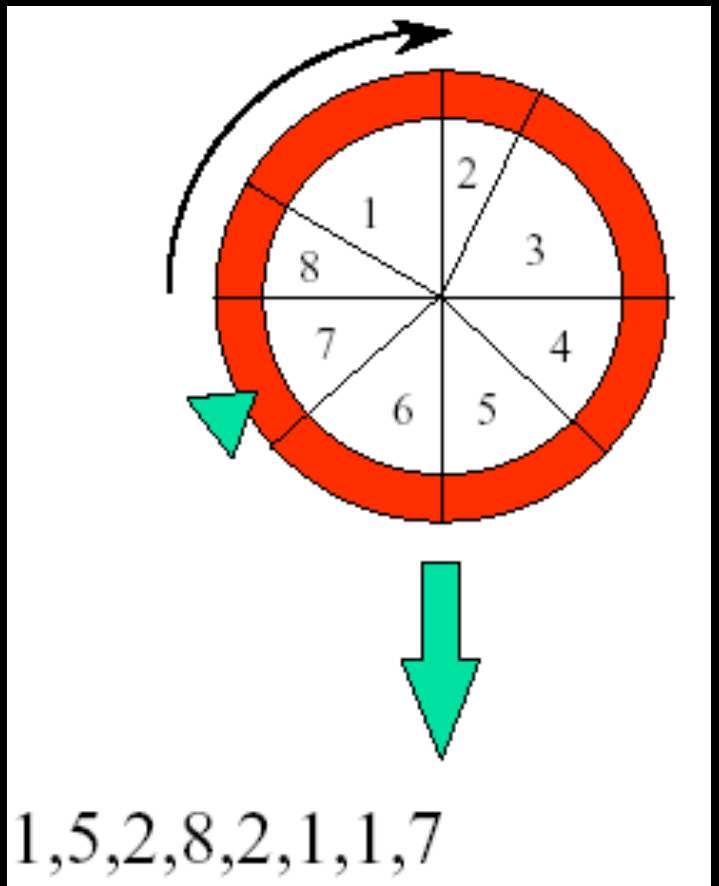
  - **Somme des adaptations**

    $5 + 20 + 11 + 8 + 6 = 50$

  - **Calcul des probabilités**
  - $Adaptation(i) / 50$

Reproduction

- Sélectionne les bonnes chaînes d'une population et forme un vivier d'accouplement
- Une simulation de roulette sert à la sélection



Techniques de sélection

- Encombrement
- Un problème potentiel lié à la sélection
  - Se produit quand les individus les plus aptes se reproduisent rapidement, si bien qu'une grande part de la population se ressemble
  - Réduit la diversité de la population
  - Peut freiner la progression de l'algorithme à long terme

Croisement

- Croisement pour les représentations vectorielles :
  - Choisir une ou plusieurs paires d'individus comme parents et échanger aléatoirement leurs segments
  - Aussi appelé « couper-coller »

- Paramètres
  - Taux de croisement
  - Nombre de points de croisement
  - Position des points de croisement

Croisement – suite

- Croisement en un point

  (croisement à 1 point)
  - Choisir un point de coupure dans les vecteurs des parents et échanger les segments

- Par exemple
  - Parents :

    1101101101 et 0001001000
  - Point de croisement : après le 4e chiffre

    1101101101 et 0001001000
  - Enfants produits :

    1101001000 et 0001101101

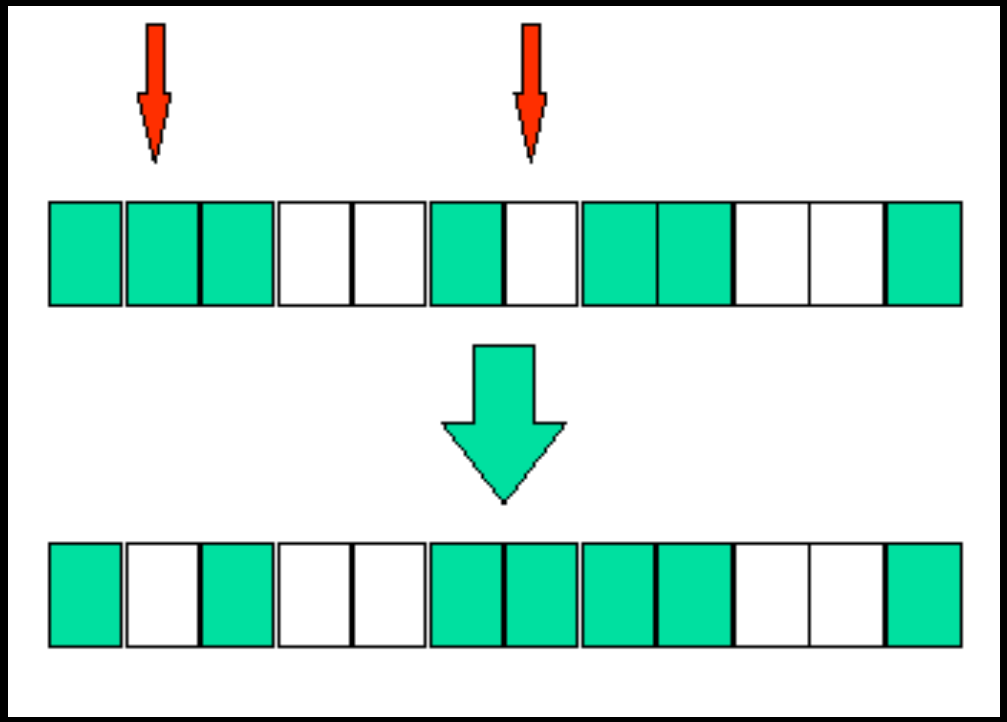
Mutation

- Modifier aléatoirement un individu

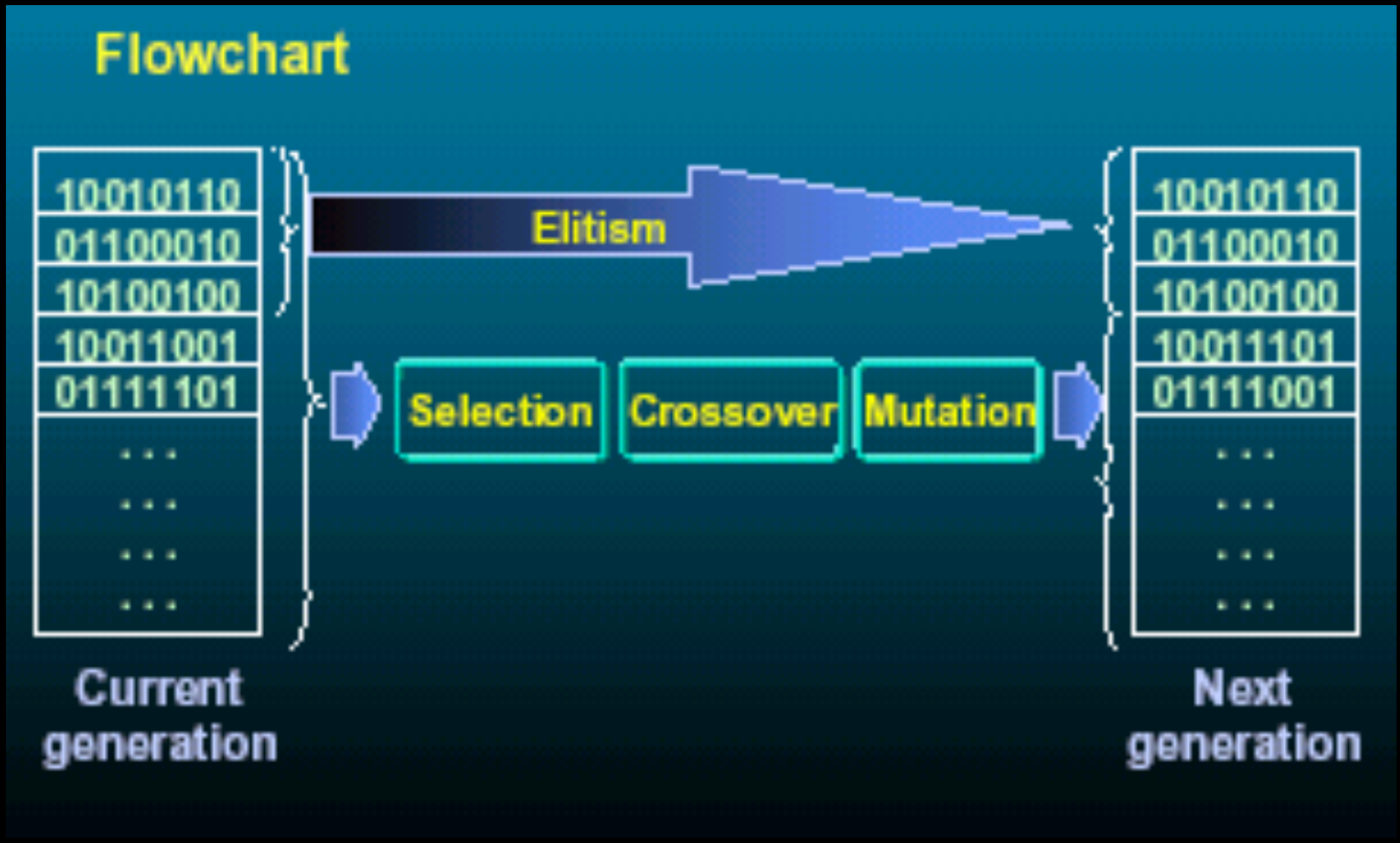


- La mutation bit à bit se fait en inversant chaque bit avec une probabilité p_m
- Utilité
  - Créer un point au voisinage du point courant pour une recherche locale
  - Maintenir la diversité de la population
- Paramètres
  - Taux de mutation
  - Ampleur de la mutation

Mutation



Résumé des opérations d'un AG

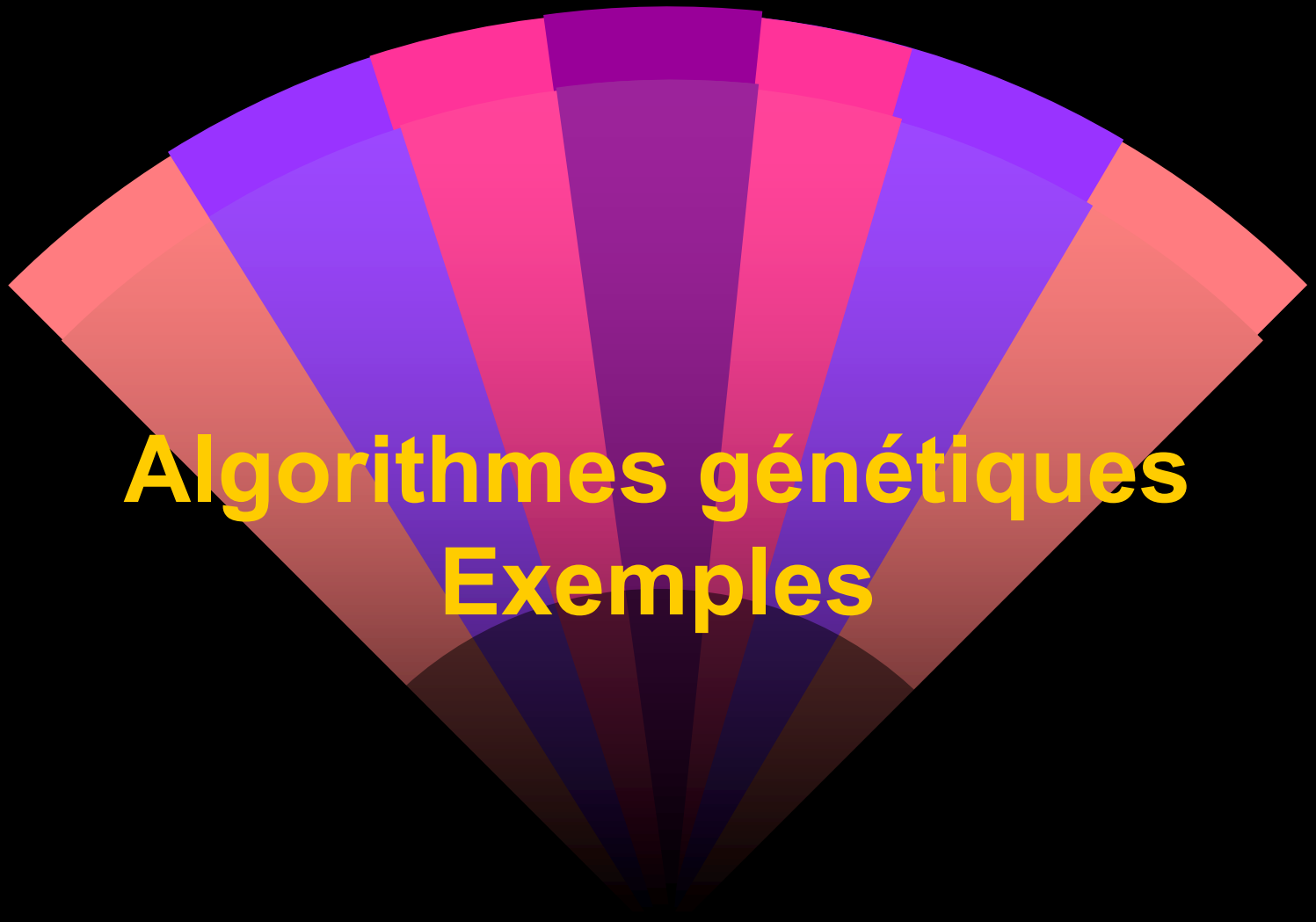


Résumé des opérations d'un AG – suite

- Les trois opérateurs sont simples et directs
- L'opérateur de reproduction sélectionne les bonnes chaînes
- L'opérateur de croisement recombine de bonnes sous-chaînes issues de bonnes chaînes pour, espérons-le, créer une meilleure sous-chaîne
- L'opérateur de mutation modifie localement une chaîne pour, espérons-le, créer une meilleure chaîne
- Même si rien de cela n'est garanti ni vérifié lors de la création d'une chaîne, on s'attend à ce que les mauvaises chaînes soient éliminées par l'opérateur de reproduction à la génération suivante

Algorithme génétique - pseudo-code

```
1:  begin
2:  define problem dimension (D), generate initial population P_0 of size NP
3:  evaluate the fitness of NP individuals in P_0
4:  initialize generation counter t ← 0
5:  while termination criterion not satisfied do
6:  |   selection of parents on the basis of their fitness
7:  |   create offspring for the new generation P_{t+1} from the parents using
8:  |  crossover operation
9:  |   perform mutation on new generation P_{t+1}
10: |   evaluate the fitness of population P_{t+1}
    |   t ← t+1
11:     end
12: end
```



Algorithmes génétiques
Exemples

Problème 1

Trouver le minimum de la fonction

(fonction de Himmelblau) (sur l'intervalle 0 à 6)

$f(x_1,x_2) = (x_1^2+x_2-11)^2 + (x_1+x_2^2-7)^2$

Solution exacte (3,2)

Solution obtenue par l'AG

La fonction est minimale au point

(2,997, 2,029) et la valeur est 0,013

Paramètres de l'AG - étape 1

Nombre de fonctions objectif	1
Fonction objectif	Minimiser
Type de croisement	AG binaire (un point)
Stratégie	1 site de croisement avec échange
Taille de la population	20
Nombre total de générations	30
Probabilité de croisement	0,8
Probabilité de mutation	0,05
Graine aléatoire saisie	0,76
Longueur de chaîne	20
Nombre de variables binaires	2
Epsilon de proximité	0
Valeur sigma-share	0,1500
Stratégie de partage	partage dans l'espace des paramètres

Bornes inférieure et supérieure

0 <= x1 <= 6

0 <= x2 <= 6

Procédure

- Codage binaire
  - Avec 10 bits, on obtient une précision de (6-0)/(2^{10}-1), soit 0,006 sur l'intervalle (0,6).
- Étape 2
  - Évaluer chaque chaîne de la population
  - La sous-chaîne 1010010001 se décode en (2^9+2^7+2^4+2^0)=657. La valeur du paramètre correspondant est 0+(6-0)*657/1023 = 3,853
  - Pour la minimisation : F(X)=1/(1+f(X))

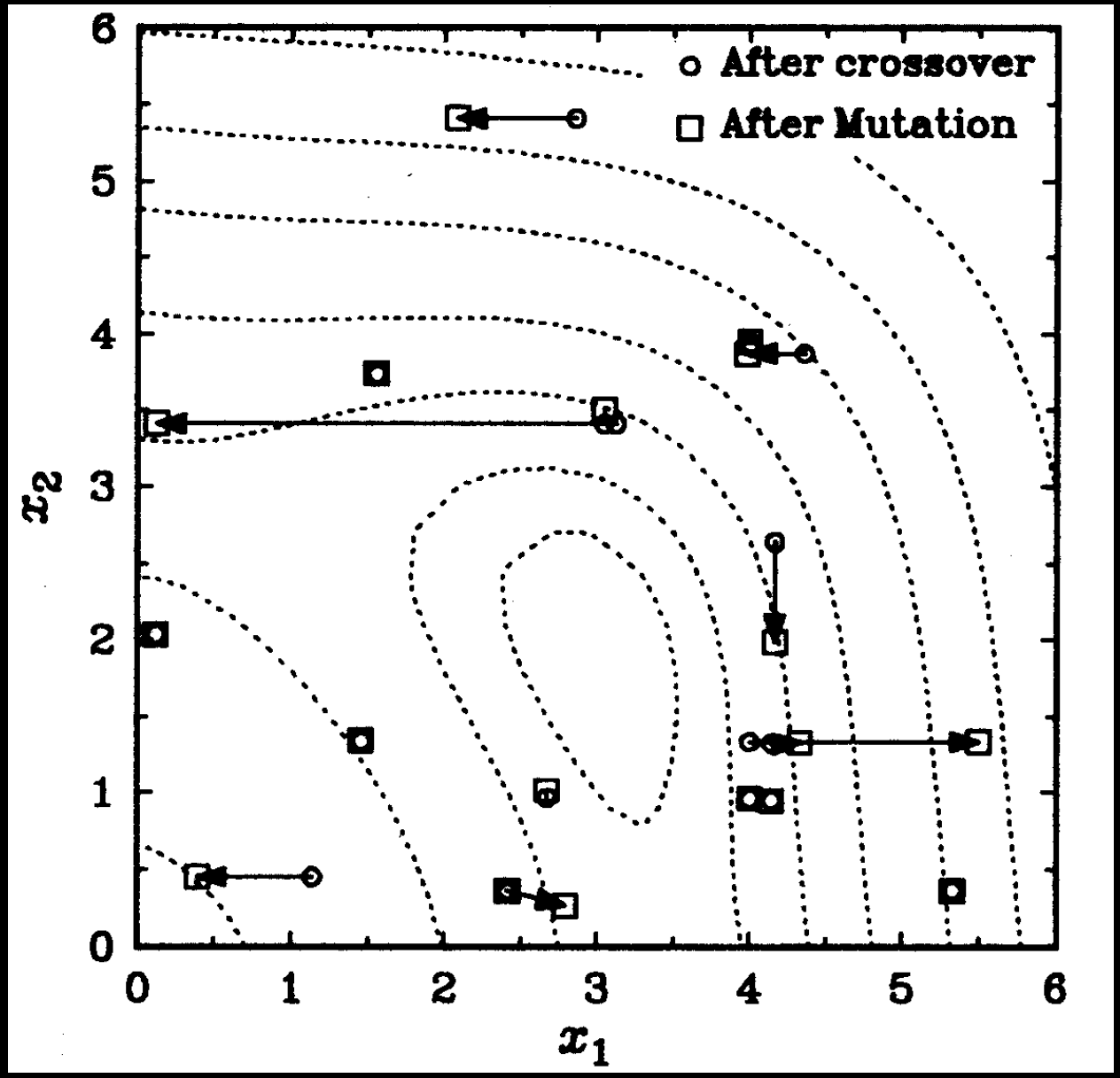
Opérations de l'AG (étapes 4-6)

Mating pool				Intermediate population		Mutation					
Substring-2	Substring-1	G	H	Substring-2	Substring-1	Substring-2	Substring-1	x_1	x_2	$f(x)$	$\mathcal{F}(x)$
0010100100	1010101010	Y	9	0010100101	0111001000	0010101101	0111001000	1.015	2.674	18.886	0.050
1010100001	0111001000	Y	9	1010100000	1010101010	1010100001	1010101010	3.947	4.000	238.322	0.004
0001001101	0011100111	Y	12	0001001101	0011000010	0001001101	0001000010	0.452	0.387	149.204	0.007
1110011011	0111000010	Y	12	1110011011	0111100111	1110011011	0101100011	5.413	2.082	596.340	0.002
0010100100	1010101010	Y	5	0010100010	1011000011	0010100010	1011000011	0.950	4.147	54.851	0.018
0011100010	1011000011	Y	5	0011100100	1010101010	0011100100	1110101010	1.337	5.501	424.583	0.002
0011100010	1011000011	N		0011100010	1011000011	0011100011	1011100011	1.331	4.334	83.929	0.012
0111000010	1011000110	N		0111000010	1011000110	0110010010	1011000110	1.982	4.164	70.472	0.014
0101011011	0000000111	Y	14	0101011011	0000010100	0101011011	0000010100	2.035	0.117	87.633	0.011
1001000110	1000010100	Y	14	1001000110	1000000111	1001010110	1000000111	3.507	3.044	72.789	0.014
0011100101	0011111000	Y	1	0011100101	0011111000	0011100101	0011111000	1.343	1.455	70.868	0.014
0011100101	0011111000	Y	1	0011100101	0011111000	0011100101	0011111000	1.343	1.455	70.868	0.014
0000111101	0110011101	N		0000111101	0110011101	0000101101	0111011100	0.264	2.792	25.783	0.037
0000111110	1110001101	N		0000111110	1110001101	0000111110	1110001101	0.364	5.331	318.746	0.003
0000111101	0110011101	Y	18	0000111101	0110011100	0000111101	0110011100	0.358	2.416	42.922	0.023
1001000110	1000010100	Y	18	1001000110	1000010101	1001000110	0000010101	3.413	0.123	80.127	0.012
1001111101	1011100111	Y	10	1001111101	0100001001	1001111101	0100001001	3.736	1.554	95.968	0.010
1010010100	0100001001	Y	10	1010010100	1011100111	1010010100	1010100111	3.871	3.982	219.426	0.005
0000111101	0110011101	N		0000111101	0110011101	0000111101	0110011101	0.358	2.422	42.598	0.023
0010100100	1010101010	N		0010100100	1010101010	0010100100	1010101010	0.962	4.000	39.849	0.024

G : Whether crossover (Y yes, N no), H : Crossing site

Source : Deb, 2002

Population après mutation



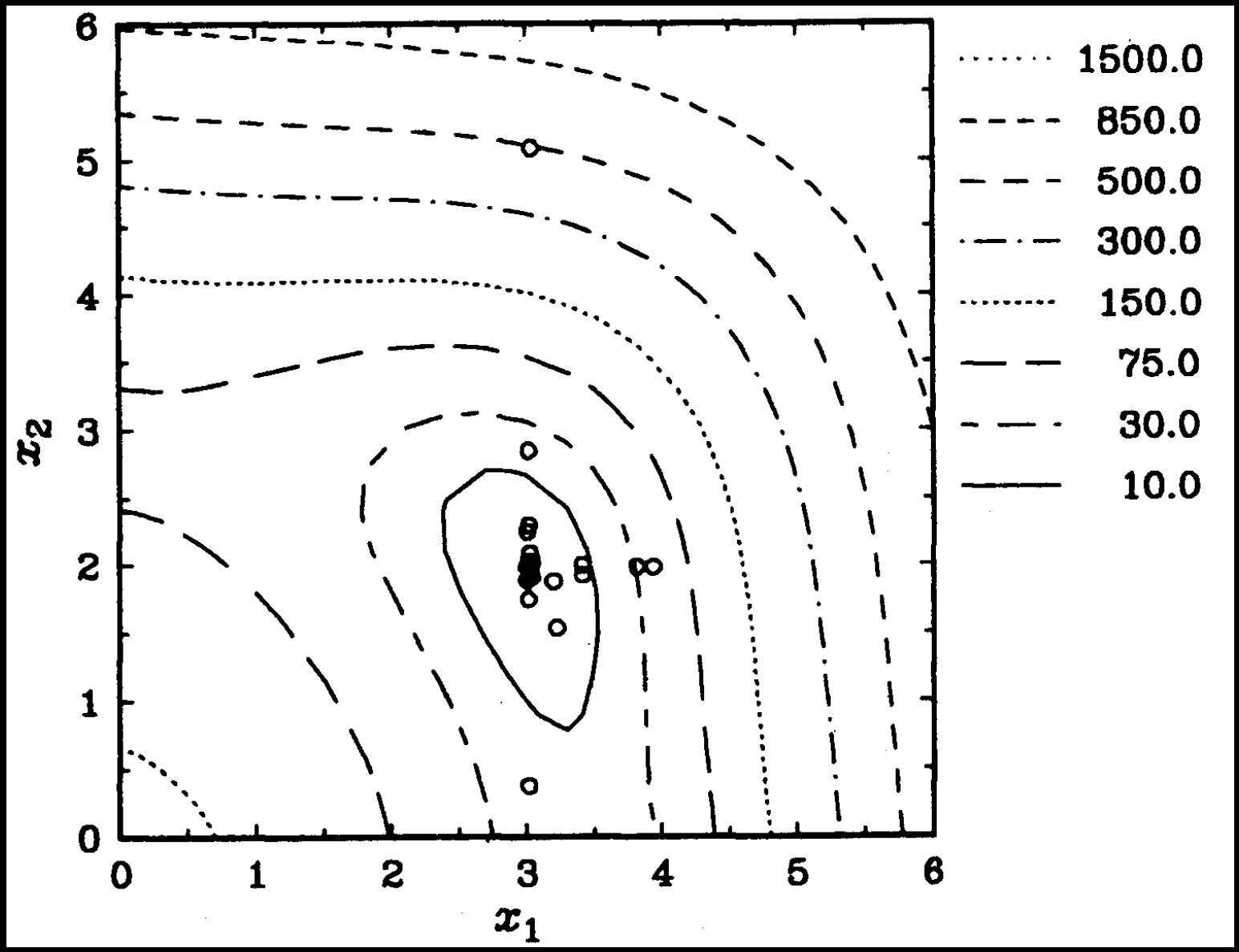
Résultats de l'AG à la 1re génération

Taille de la population	Valeurs de x_1	Valeurs de x_2	f(x) Valeurs	Chaîne
1	3.85	1.84	32.44	1010010001_0100111010
2	0.70	1.31	105.36	0001111000_0011100000
3	1031	5.89	1661.14	1110010100_1111101101
4	5.37	4.45	345.85	1010010000_1011110111
5	3.85	5.15	691.48	1011000101_1101101110
6	1.54	3.39	63.90	0100000110_1001000010
7	4.02	3.52	164.26	1010101101_1001011001
8	3.53	4.96	379.97	0100000000_1101001110
9	2.05	2.93	28.13	0101011101_0111110011
10	0.91	5.64	679.93	0010011011_1111000001
11	0.03	3.11	69.61	0000000101_1000010011
12	5.44	0.54	366.43	1110011111_0001011100
13	2.48	3.54	65.52	0110100110_1001011011
14	3.15	0.28	14.89	1000011001_0000101111
15	4.60	5.77	1212.63	1100010010_1111011000
16	0.41	2.92	66.31	0001000110_0111110001
17	5.54	0.72	416.10	1110110000_0001111011
18	0.18	4.50	223.24	0000011111_1100000000
19	3.81	4.89	500.10	1010001001_1101000010
20	2.92	2.60	7.16	0111110010_0110111011

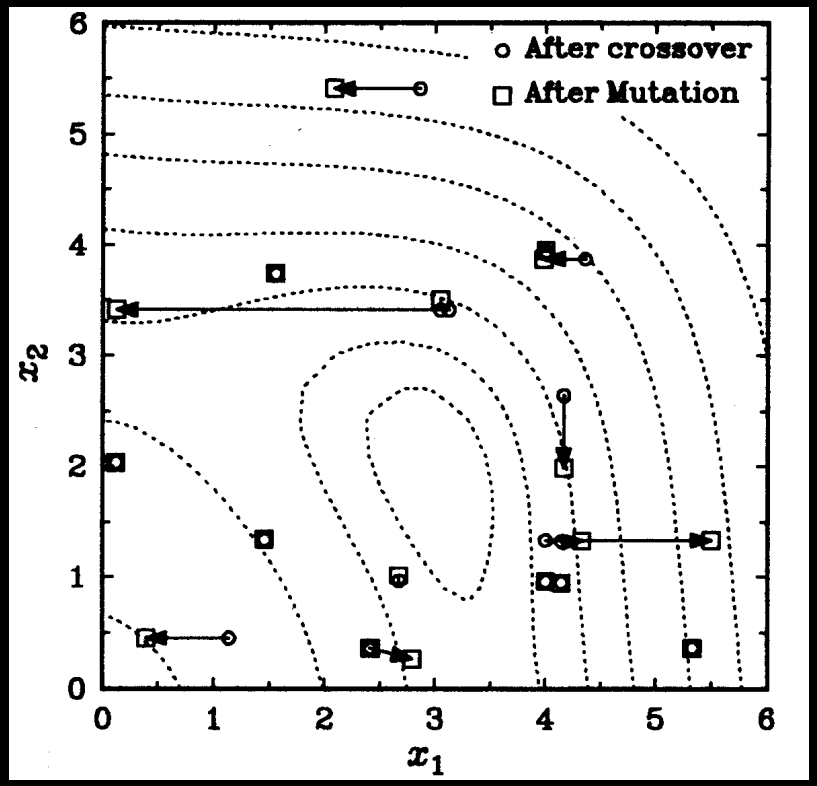
Résultats de l'AG après la 30e génération

Taille de la population	Valeurs de X_1	Valeurs de X_2	f(x) Valeurs	Chaîne
1	5.81	0.58	543.42	1111011110_0001100011
2	2.99	2.09	0.124	0111111110_0101100100
3	2.92	1.56	3.53	0111110010_0100001010
4	2.30	1.95	0.05	0111111111_0101001100
5	2.30	2.03	0.01	0111111111_0101011010
6	2.05	2.45	20.11	0101011101_0110100001
7	2.78	1.89	2.34	0111011010_0101000010
8	2.00	2.12	0.24	0111111111_0101101001
9	2.42	1.92	11.26	0110011100_0101001000
10	2.95	5.46	670.58	0111110111_1110100010
11	2.94	2.13	0.28	0111110101_0101101011
12	2.80	1.91	1.84	0111011110_0101000101
13	2.94	2.10	0.17	0111110101_0101100100
14	3.00	2.12	0.23	0111111110_0101101001
15	2.77	2.07	1.54	0111011001_0101100000
16	2.75	1.72	4.55	0111010101_0100100110
17	2.41	2.10	9.58	0110011011_0101100110
18	2.76	5.90	940.77	0111010110_1111101110
19	2.96	2.11	0.19	0111111001_0101101000
20	2.30	2.14	0.35	0111111111_0101101101

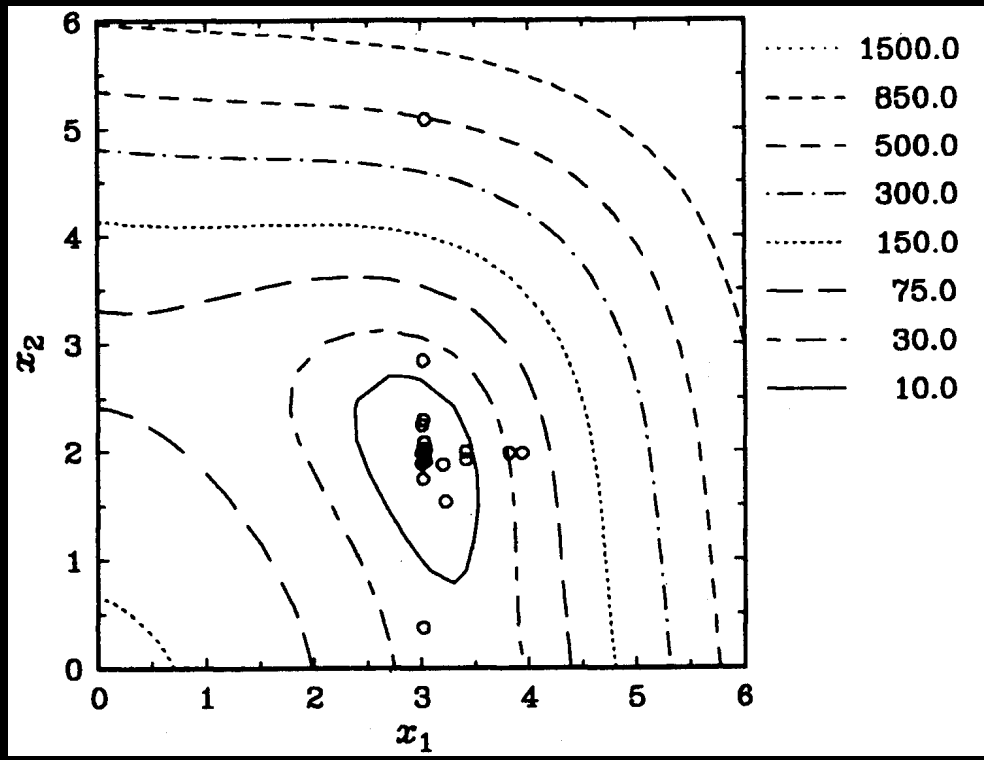
Population après la 30^{e} génération



Population après la 1re génération



Population après la 30e génération



Source : Deb, 2002

Résultats de l'AG

Après la génération 1

Adaptation max. = 1661,135

Adaptation min. = 7,155 Adaptation moy. = 354,719

Nb de mutations = 0 ; nb de croisements = 0

Après la génération 30

Adaptation max. = 940,774

Adaptation min. = 0,013 Adaptation moy. = 110,557

Nb de mutations = 20 ; nb de croisements = 245

Problème 2

- Trouver le maximum de la fonction « peaks »

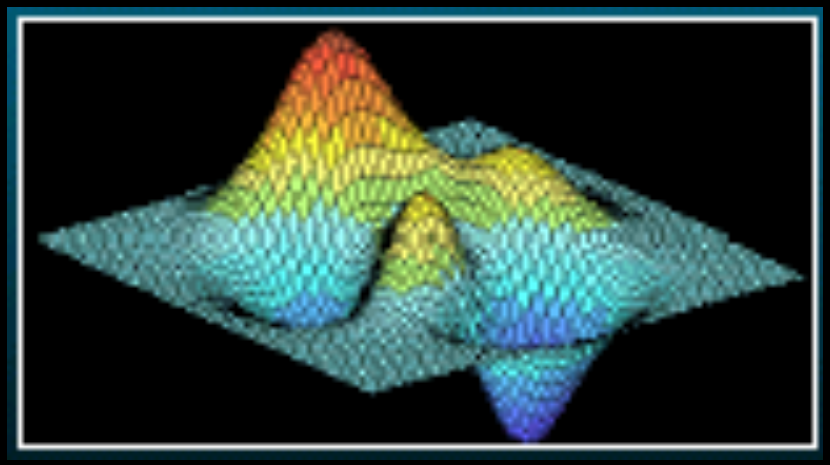
$z = f(x, y)$

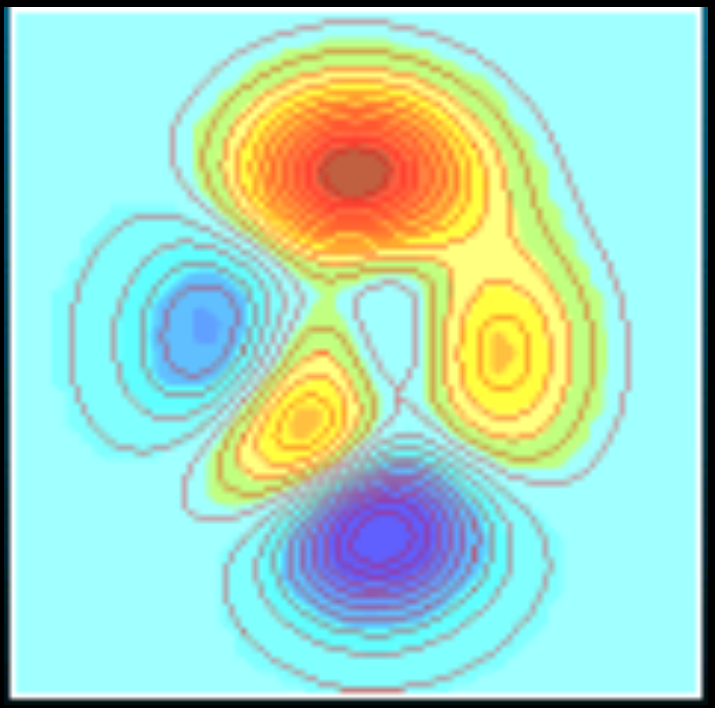
$= 3\,(1-x)^2 \exp(-x^2 - (y+1)^2)$

$- 10\,(x/5 - x^3 - y^5) \exp(-x^2-y^2)$

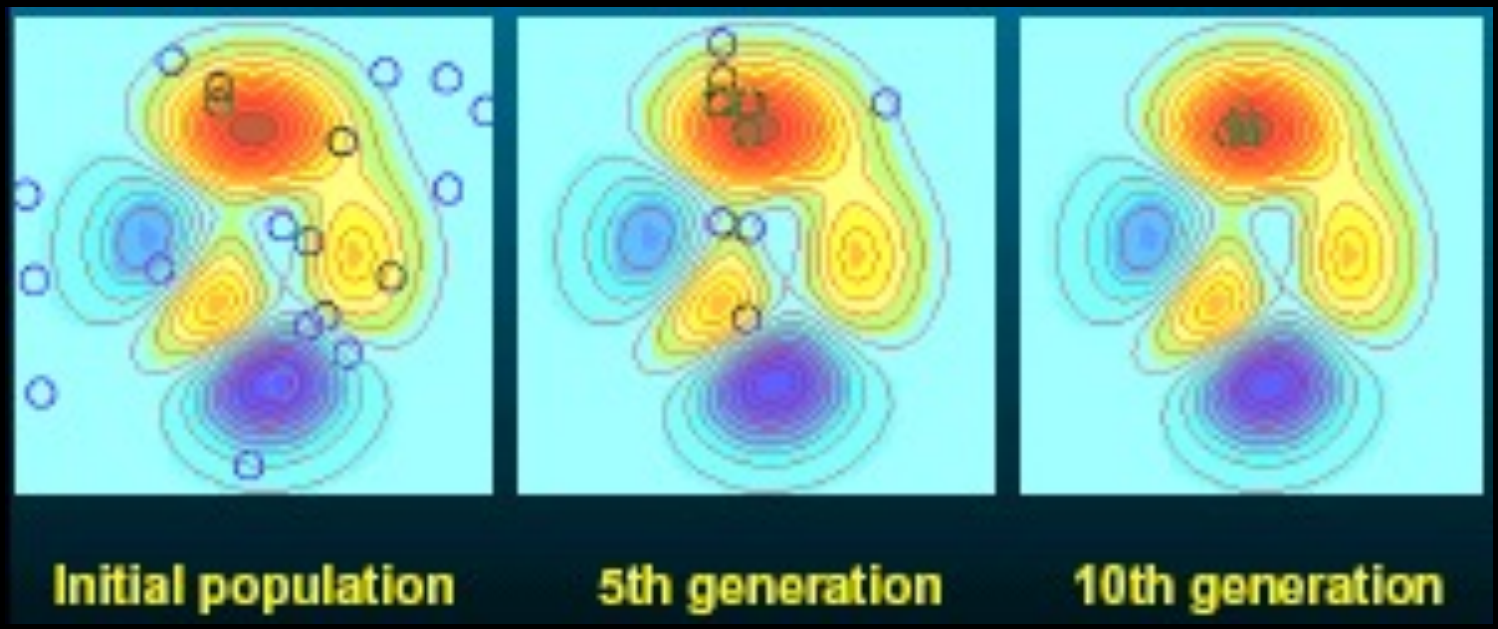
$-1/3 \exp(-(x+1)^2 - y^2)$

Tracé de la fonction « peaks »





Déroulement de l'AG





Algorithmes génétiques pour l'optimisation sous contraintes

Les AG pour l'optimisation sous contraintes

- Le problème contraint est transformé en problème non contraint par la méthode des pénalités
- Une solution hors de la région admissible reçoit une pénalité très élevée

Les AG pour l'optimisation sous contraintes

- On utilise un terme de pénalité à opérateur crochet

$$F_i = f(x) + \in \sum_{j=1}^{k} \delta_j \, (\phi_j)^2$$

- F_i est la valeur d'adaptation
- $f(x)$ est la valeur de la fonction objectif
- k est le nombre total de contraintes
- $\in$ vaut -1 pour une maximisation et +1 pour une minimisation
- δ_j est le coefficient de pénalité
- ϕ_j est l'ampleur de la violation

Problème 1

Trouver le minimum de la fonction (fonction de Himmelblau)

$f(x_1,x_2) = (x_1^2+x_2-11)^2 + (x_1+x_2^2-7)^2$

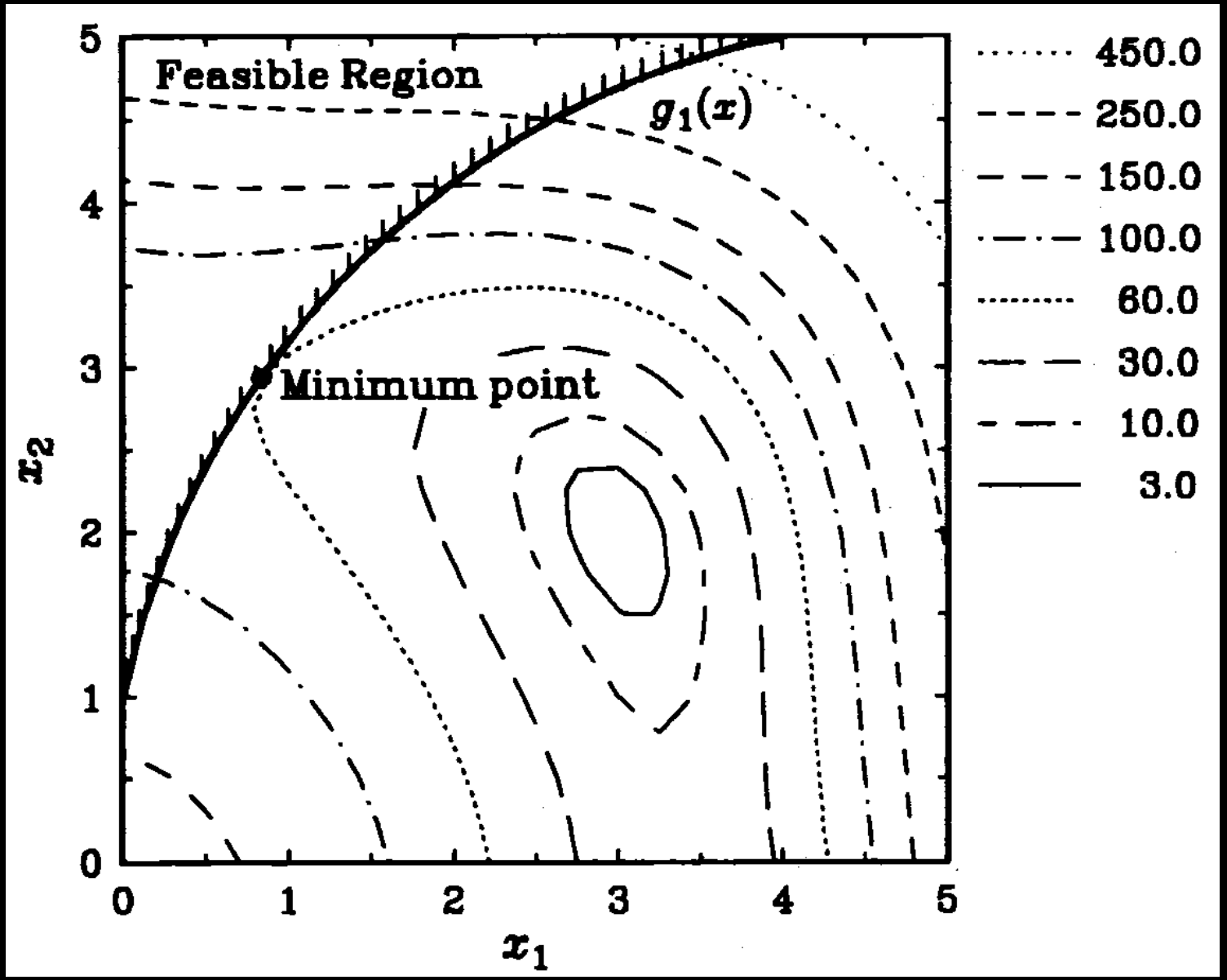
Sous les contraintes

$(x_1-5)^2 + x_2^2 - 26 \geq 0$

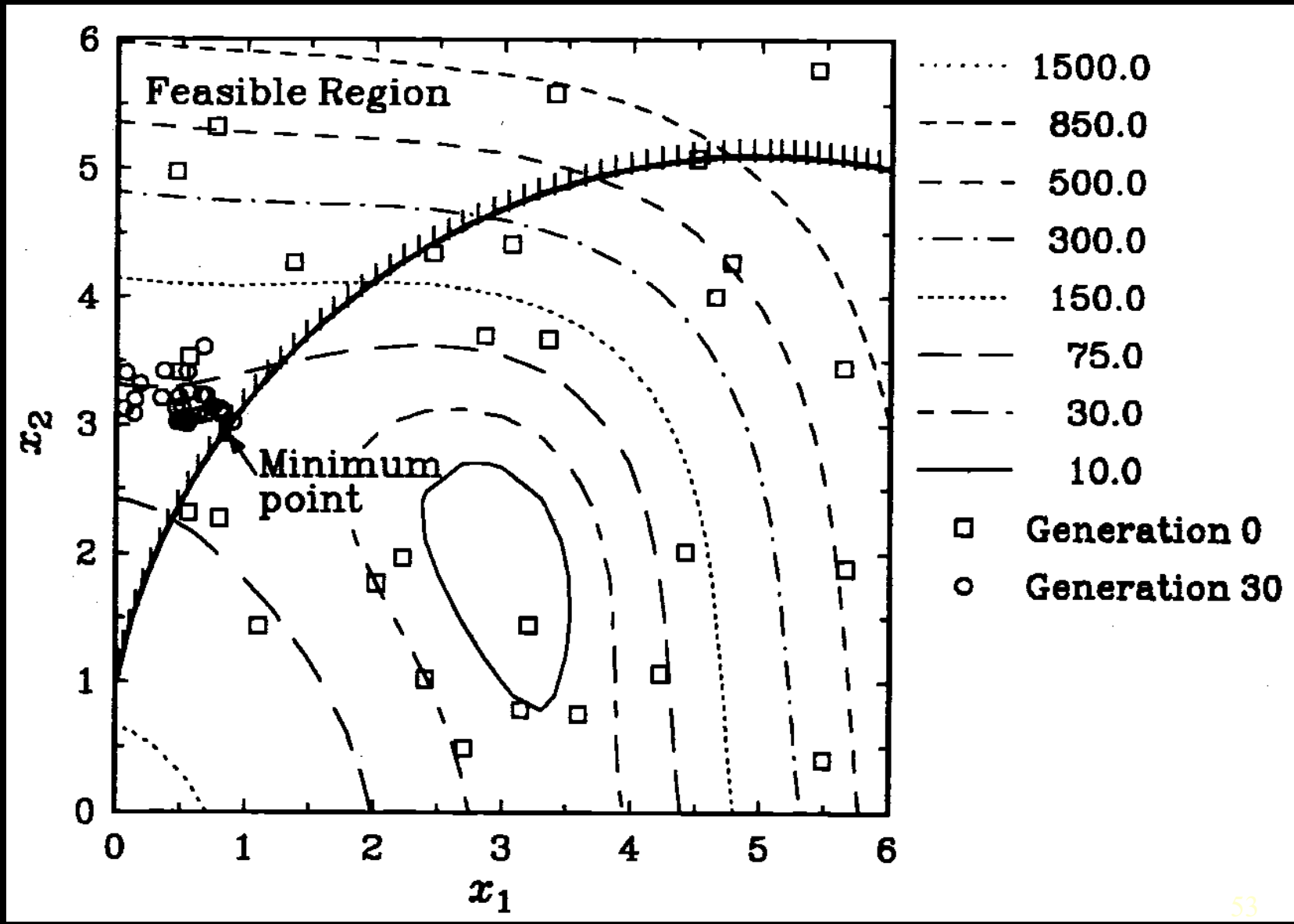
x_1 0 et x_2 0

$x_1 \leq 6$ et $x_2 \leq 6$

Solution


Feasible Region
$g_1(x)$
Minimum point
450.0
250.0
150.0
100.0
60.0
30.0
10.0
3.0
x_1
x_2
0
1
2
3
4
5

Solution par AG





Optimisation mono-objectif pour la gestion des réservoirs

Application des AG en ingénierie des ressources en eau

- Optimisation hydroélectrique d'un réservoir
  - Maximiser la production hydroélectrique annuelle, les autres demandes en eau du réservoir étant des contraintes
  - Un pas de temps mensuel est utilisé pour démontrer l'applicabilité des AG
    - 12 périodes dans l'année

Description de la zone d'étude

- **Barrage de Hirakud : un projet polyvalent sur le fleuve Mahanadi, État d'Odisha, Inde**
  - **Maîtrise des crues, irrigation et hydroélectricité**
- **Situation : lat. 21° 32' N ; long. 83° 52' E**
- **Bassin versant jusqu'au barrage : 83 400 km²**
- **Capacité de stockage utile : 7 189 millions de m³**
- **Puissance installée des turbines : 307,5 MW**
- **Superficie irriguée**
  - **Première saison de culture, *Kharif* : 1 565,5 km²**
  - **Seconde saison de culture, *Rabi* : 1 084,0 km²**

Problème d'optimisation

Fonction objectif

$$Maximize Z = \sum_{t=1}^{12} P_t$$

$P_t = 2725\ RP_t\ h_t$

RP - eau lâchée pour la production électrique (millions de m^3)

h - charge en mètres

Contraintes

- Lâchers pour la puissance garantie et capacité des turbines

$$RP_t \leq TC \quad \forall \quad t = 1,2,.....12$$

$$RP_t \geq FP_t \quad \forall \quad t = 1,2,.....12$$

- Contraintes de demande d'irrigation

$$RI_t \geq ID_{MIN_t} \quad \forall \quad t = 1,2,.....12$$

$$RI_t \leq ID_{MAX_t} \quad \forall \quad t = 1,2,.....12$$

Contraintes – suite

- Contraintes de continuité du stockage du réservoir

$$(1+a_t)S_{t+1} = (1-a_t)S_t + Q_t - RI_t - RP_t - OVF_t - A_o e_t \quad \forall \text{ t}$$

- Contraintes de continuité du stockage du réservoir

$$S_t \leq S_{\max} \qquad \forall \; t = 1,2,.....12$$

Mise en œuvre dans l'AG

- Identification des variables de décision
  - Lâcher pour la production électrique chaque mois (12)
- Fixation des bornes inférieure et supérieure
  - La borne inférieure est la puissance garantie
  - La borne supérieure est la capacité des turbines
- Fixation de la longueur de la chaîne binaire
  - Selon l'écart entre les bornes inférieure et supérieure de chaque variable de décision
- Codage de la chaîne
  - Le codage binaire se fait en générant des nombres aléatoires

Mise en œuvre dans l'AG – suite

- Décodage des variables de décision
  - Règle de correspondance linéaire
- Calcul de l'adaptation
  - Les valeurs des variables de décision et la charge effective sont introduites dans la fonction d'adaptation pour évaluer chaque chaîne
- Opérations de l'AG
  - Les 3 étapes des opérations de l'AG, à savoir reproduction, croisement et mutation, sont appliquées aux chaînes
  - Probabilité de croisement : 0,8
  - Probabilité de mutation : 0,006
- Répéter jusqu'au critère d'arrêt

Bornes inférieure et supérieure et longueur de chaîne de la variable de décision

Month	Decision Variable	Lower bound	Upper bound	String length
January	RP_1	616.50	2000	10
February	RP_2	370.00	2000	15
March	RP_3	370.00	2000	15
April	RP_4	245.00	2000	15
May	RP_5	370.00	2000	15
June	RP_6	370.00	2000	15
July	RP_7	615.00	2000	10
August	RP_8	1233.00	2000	10
September	RP_9	1110.00	2000	10
October	RP_{10}	615.00	2000	10
November	RP_{11}	615.00	2000	10
December	RP_{12}	615.00	2000	10

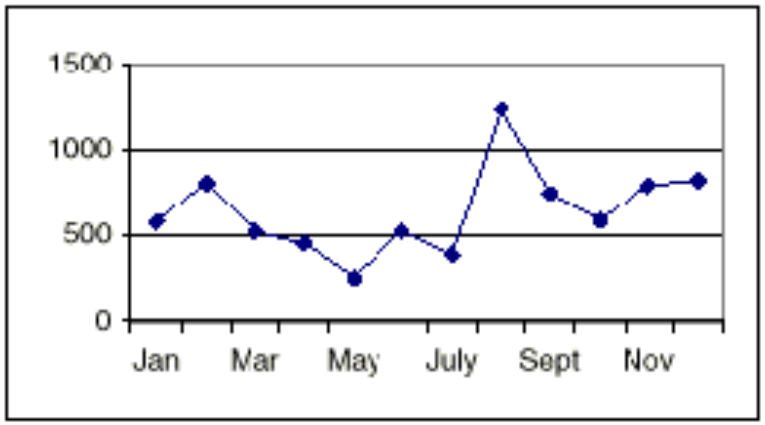
Apports moyens et demandes d'irrigation selon les mois

Month	Inflows M.cu.m.	Irrigation Demand M.cu.m
January	397.026	229.831
February	178.785	268.671
March	33.291	323.046
April	46.854	313.428
May	6.165	57.334
June	431.550	107.518
July	4,151.511	249.683
August	18,377.865	243.271
September	5,611.383	265.588
October	1,392.057	295.304
November	626.364	45.005
December	556.083	117.135

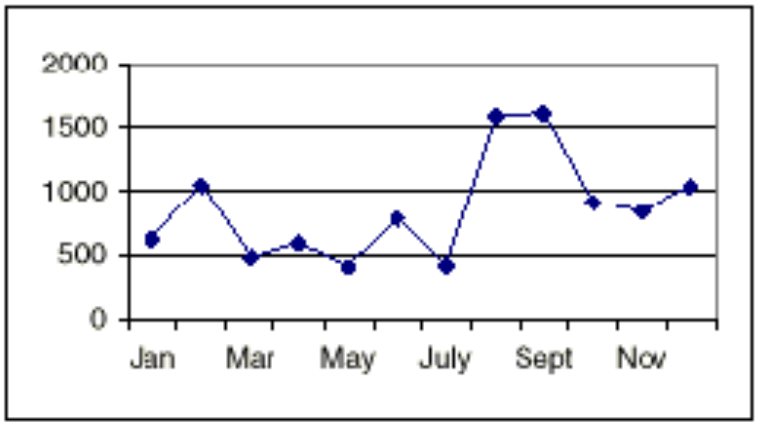
Scénarios d'apports

- Le modèle a été exécuté pour quatre niveaux d'apports garantis
  - Apports inférieurs de 40 % et de 20 % à la moyenne
  - Apports moyens et supérieurs de 20 % à la moyenne

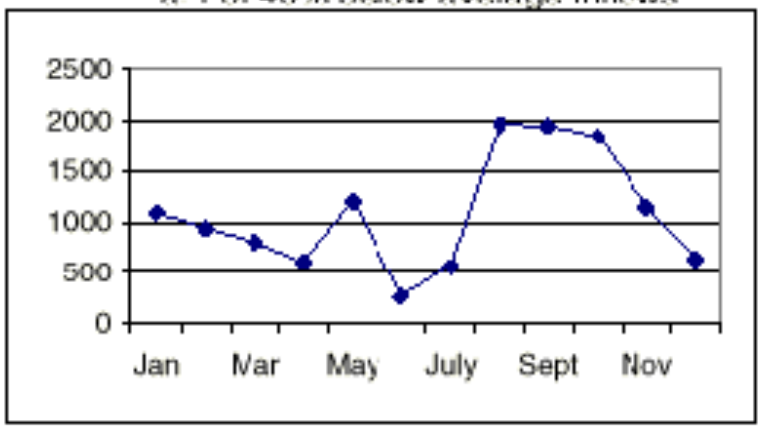
Lâchers optimaux pour la production électrique (Mm^3)



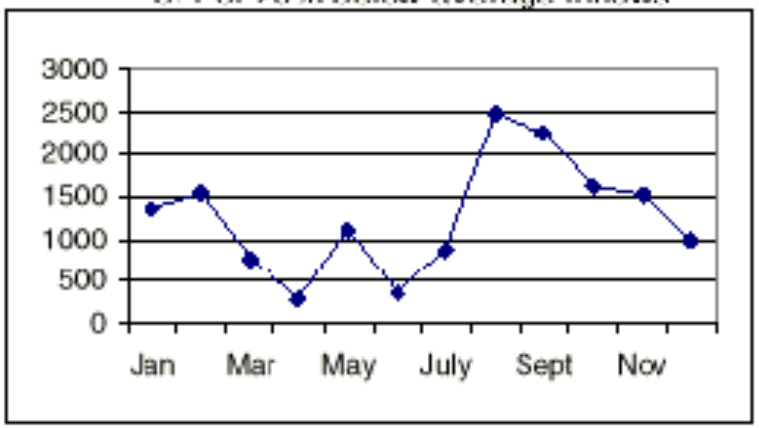
a. For 40% below average inflows



b. For 20% below average inflows



c. For average inflows



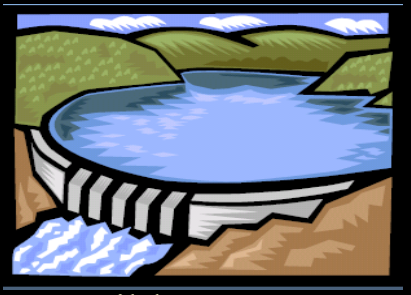
d. For 20% above average inflows



Optimisation multi-objectif

Objectifs multiples dans la gestion des réservoirs

- ***Un système de ressources en eau***
  - *implique des objectifs contradictoires*
  - *implique de nombreuses variables qui influent de façon non linéaire sur toutes les caractéristiques et le comportement du système*

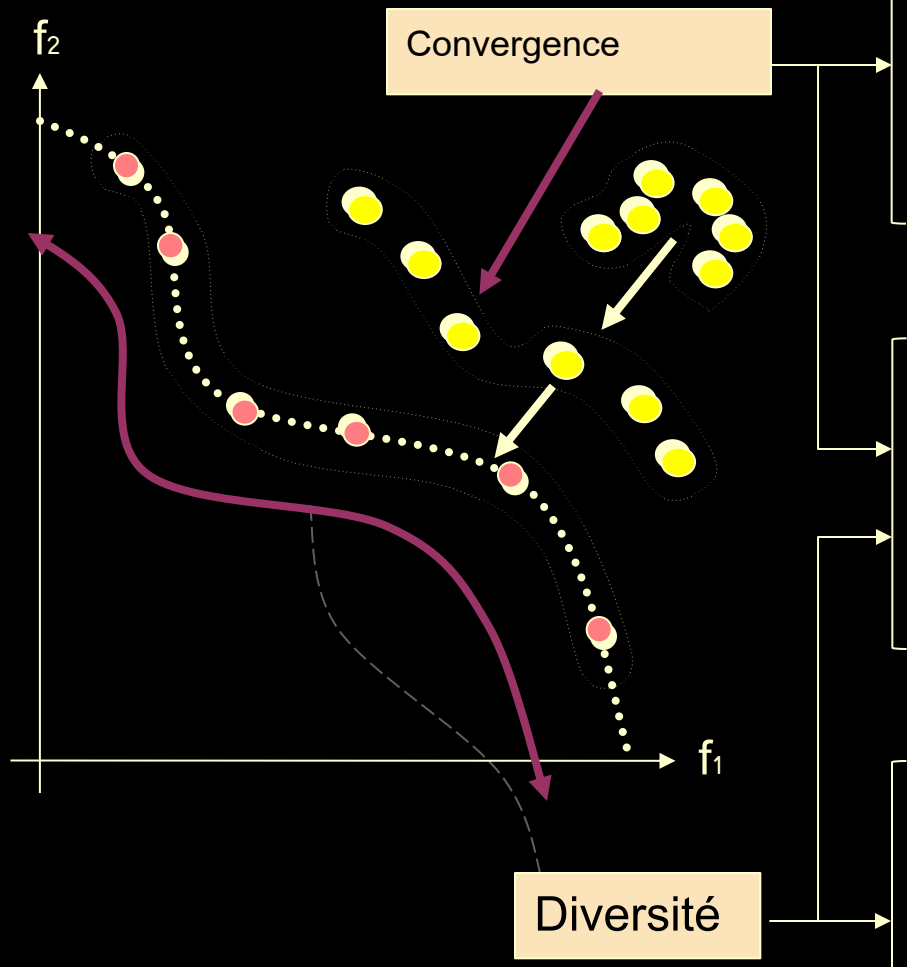


Exemples :

Gestion des réservoirs

- Minimiser le risque ***d'inondation*** (f_1) aussi proche de 0 que possible
- Maximiser la production ***hydroélectrique*** (f_2) aussi élevée que possible
- Minimiser les déficits ***d'irrigation*** (f_3) aussi faibles que possible
- Maximiser la ***qualité de l'eau*** du fleuve en aval (f_4) aussi élevée que possible

Enjeux des AEMO



- Comment guider la population vers l'ensemble de Pareto ?
  ❶ **attribution de l'adaptation**

Comment éviter de perdre les solutions non dominées ?

sélection environnementale

- Comment maintenir une approximation diversifiée de l'ensemble de Pareto ?

estimation de densité



Optimisation multi-objectif pour la gestion des réservoirs

Optimisation multi-objectif d'un système de réservoir – étude de cas

Réservoir de Bhadra – un réservoir polyvalent

- Situation - district de Chikmagalur, État du Karnataka, Inde

  75°38'20'' E de longitude et 13°42' N de latitude

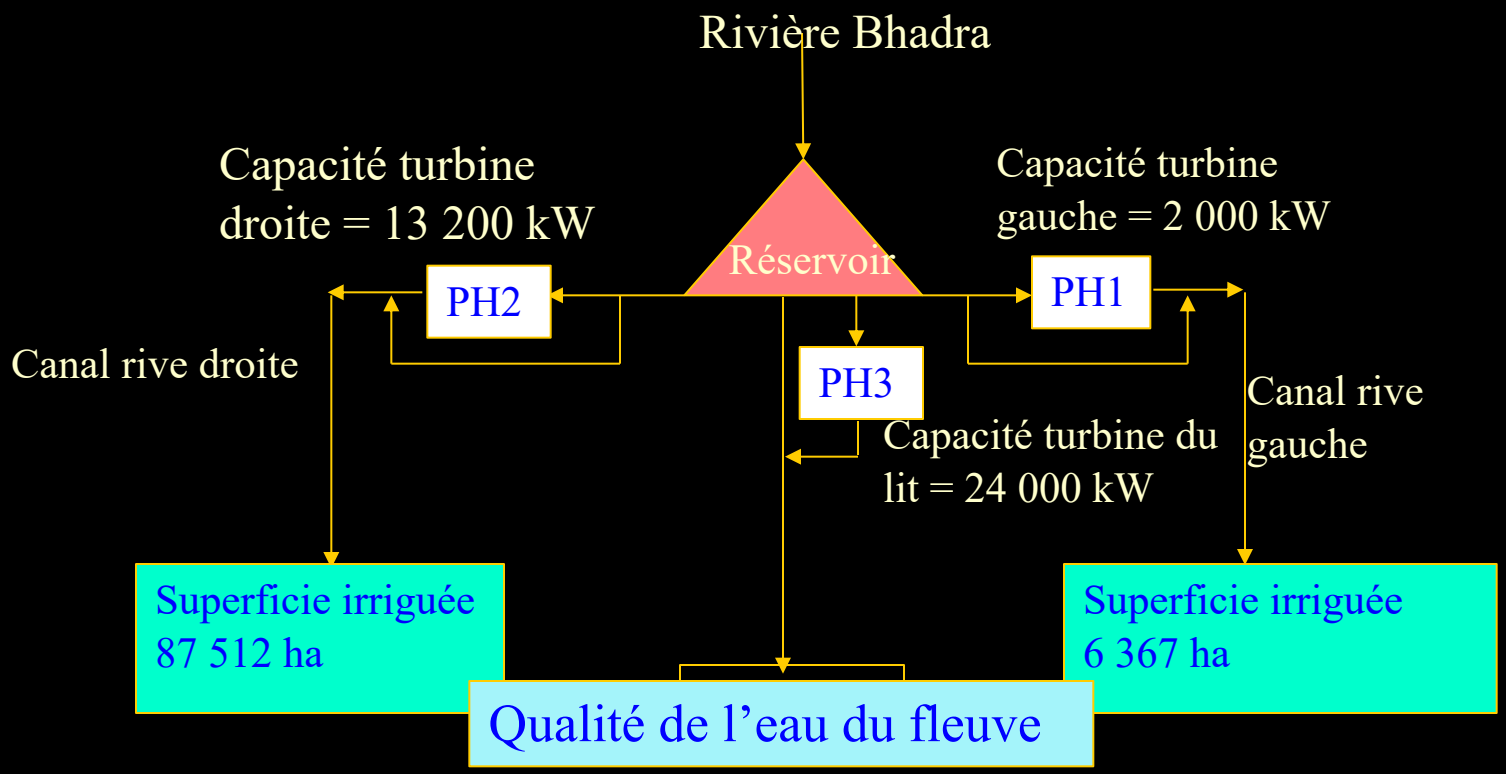


Schéma du projet de réservoir de Bhadra

Modèle multi-objectif de gestion de réservoir

1. **Minimiser**

Déficit d'irrigation (f_1) :

$$SQDV = \sum_{t=1}^{12} \left(D_{l,t} - IR_{l,t}\right)^2 + \sum_{t=1}^{12} \left(D_{r,t} - IR_{r,t}\right)^2$$

2. **Maximiser**

Production hydroélectrique (f_2) :

$$E = \sum_{t=1}^{12} (k_1 R_{l,t} H_{l,t} + k_2 R_{r,t} H_{r,t} + k_3 R_{b,t} H_{b,t})$$

Sous contraintes de

- **dynamique du système**
- **bornes de stockage**
- **limites de production électrique maximale**
- **demandes d'irrigation et**
- **exigences de qualité de l'eau**

Contraintes

Contraintes de continuité du stockage du réservoir

$$S_{t+1} = S_t + I_t - (R_{1,t} + R_{2,t} + R_{3,t} + E_t + O_t) \quad \text{pour tout t=1, 2,…,12}$$

Contraintes de bornes de stockage

$$S_{min} \le S_t \le S_{max} \quad \text{pour tout t=1, 2,…,12}$$

Limites de capacité des turbines

$$p\, R_{1,t}\, H_{1,t} \le E_{1,max}$$
$$p\, R_{r,t}\, H_{2,t} \le E_{2,max} \quad \text{pour tout t=1, 2,…,12}$$
$$p\, R_{3,t}\, H_{3,t} \le E_{3,max}$$

Limites de capacité des canaux

$$R_{1,t} \le C_{1,max}$$
$$R_{2,t} \le C_{2,max} \quad \text{pour tout t=1, 2,…,12}$$

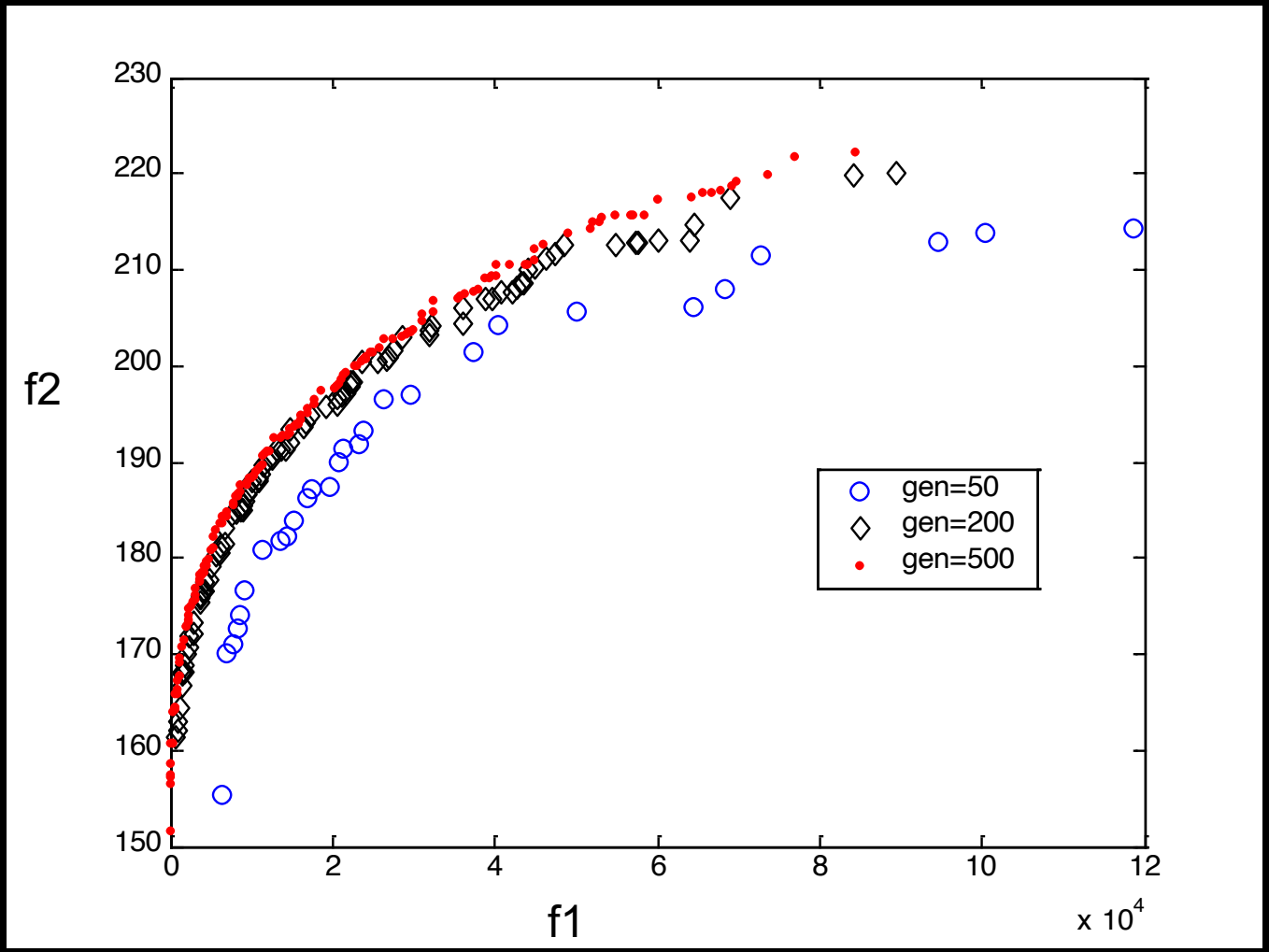
Demandes d'irrigation

$$D1_{min,\,t} \le R_{1,t} \le D1_{max,\,t}$$
$$D2_{min,\,t} \le R_{2,t} \le D2_{max,\,t} \quad \text{pour tout t=1, 2,…,NT}$$

Exigences de qualité de l'eau

$$R_{3,t} \ge MDT_t \quad \text{pour tout t=1, 2,…,NT}$$

Solution Pareto-optimale pour la gestion du réservoir



Amélioration du front de Pareto au fil des itérations.

f1 est le déficit d'irrigation annuel au carré ; f2 la production hydroélectrique en MkWh

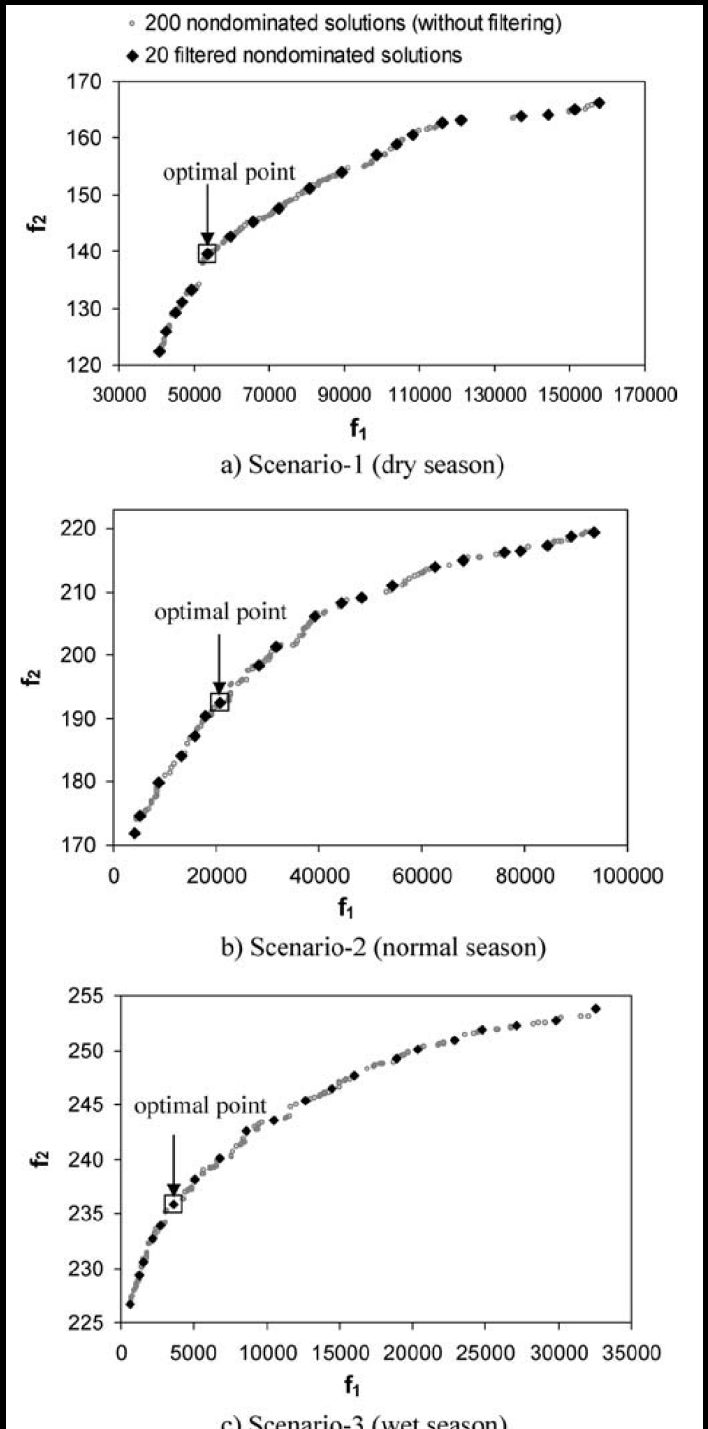
Application du modèle

Le modèle MOGA est résolu pour trois scénarios d'apports au réservoir

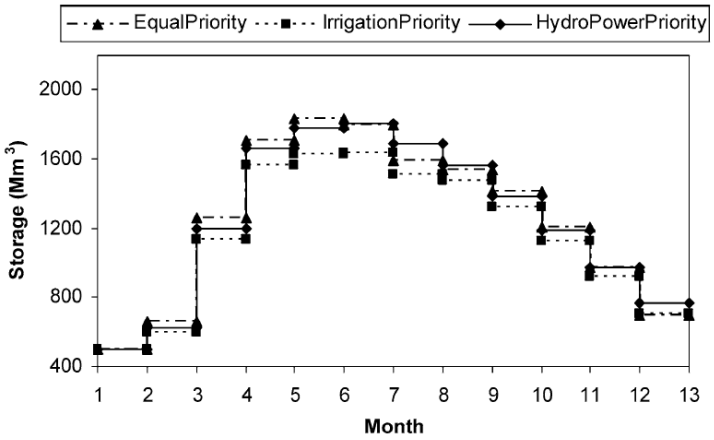
Scenario 1: Mean monthly inflows – 0.5* SD
Scenario 2: Mean monthly inflows
Scenario 3: Mean monthly inflows + 0.5* SD
where SD = standard deviation of monthly inflows.

Front de Pareto montrant le compromis entre irrigation (f_1) et hydroélectricité (f_2) pour différents scénarios d'apports.

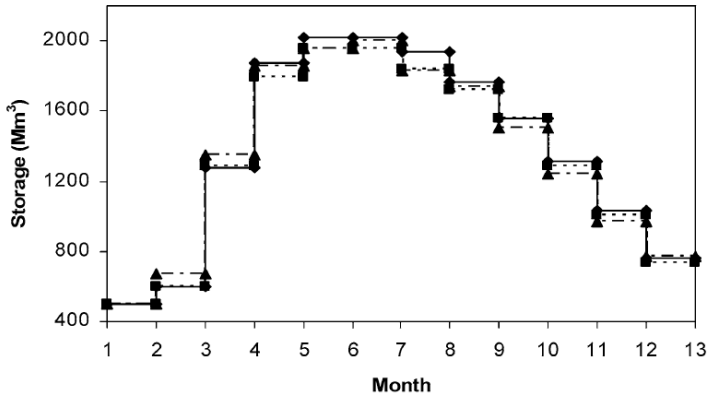
f_1 = somme des déficits d'irrigation au carré,
$(Mm^3)^2$;
f_2 = production hydroélectrique, (MkWh)



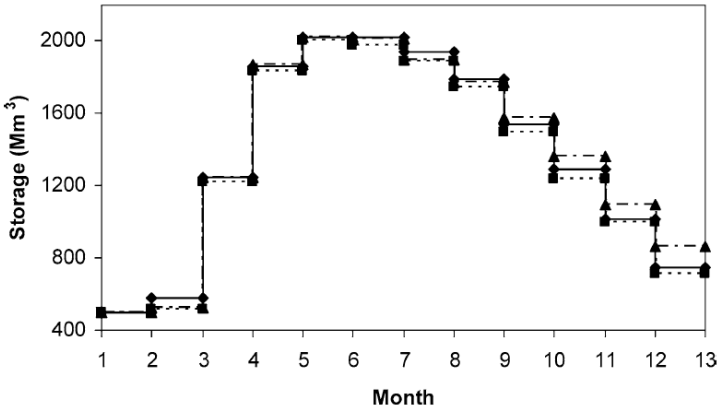
Règles de gestion du réservoir pour différents scénarios d'apports, avec les stockages initiaux selon les situations,
à savoir priorité égale, priorité à l'irrigation seule et
priorité à l'hydroélectricité seule.



a) Scenario 1 (dry season)

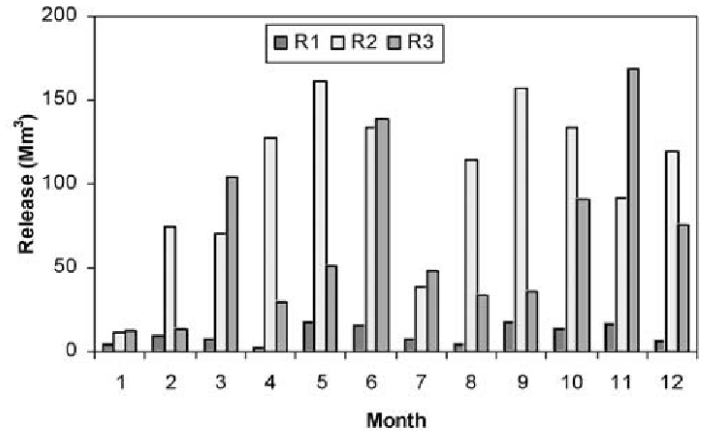


b) Scenario 2 (normal season)

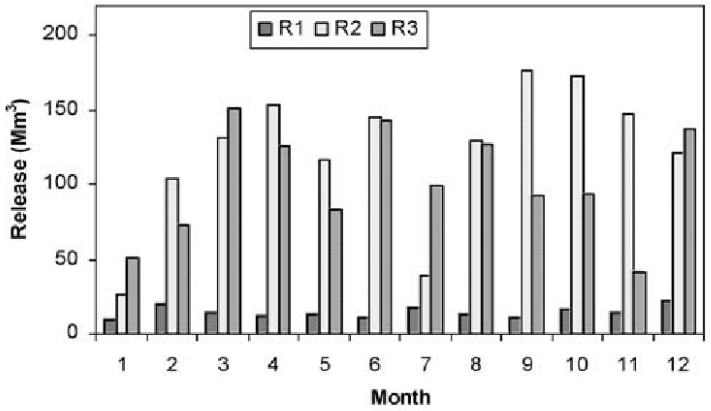


c) Scenario-3 (wet season)

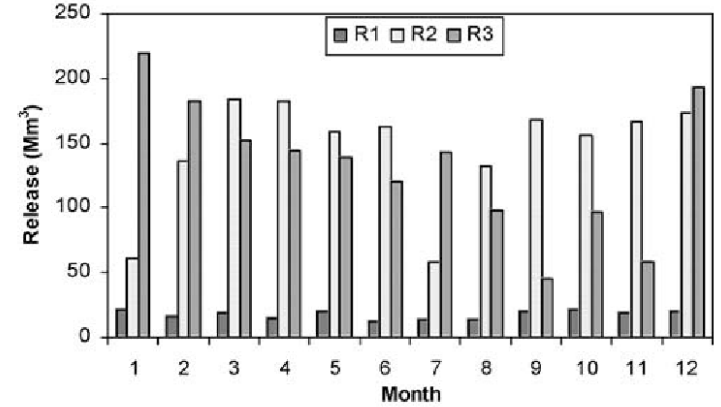
Politique de lâchers optimale pour le cas de priorité égale, avec les lâchers en Mm^3 pour le canal rive gauche (R1), le canal rive droite (R2) et le lit de la rivière (R3) selon les scénarios d'apports.



a) Scenario 1 (dry season)



b) Scenario-2 (normal season)



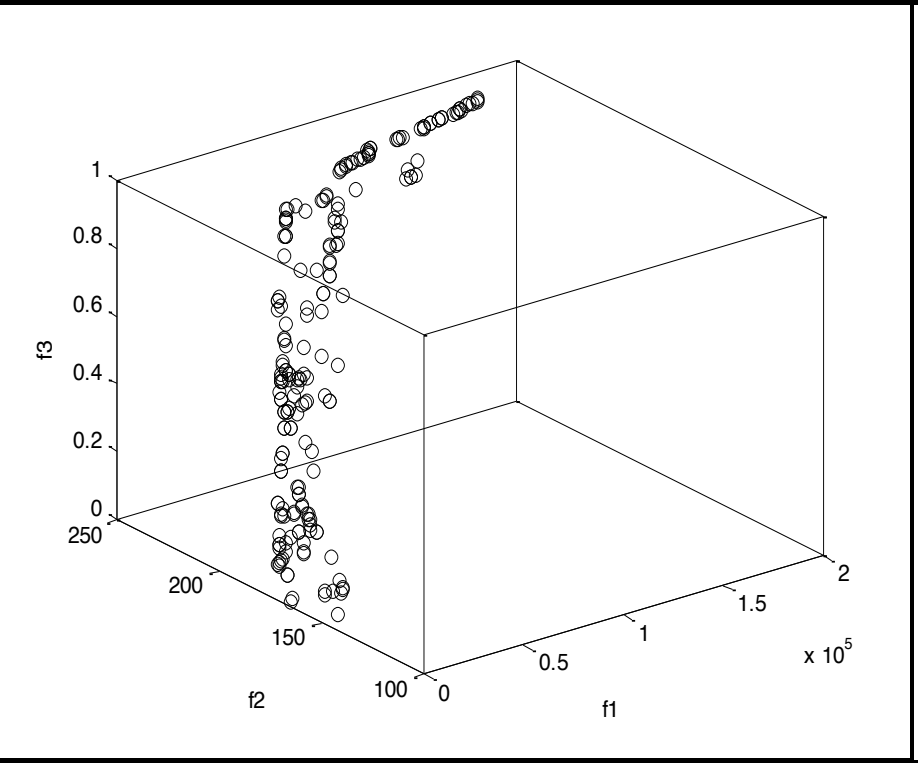
c) Scenario-3 (wet season)

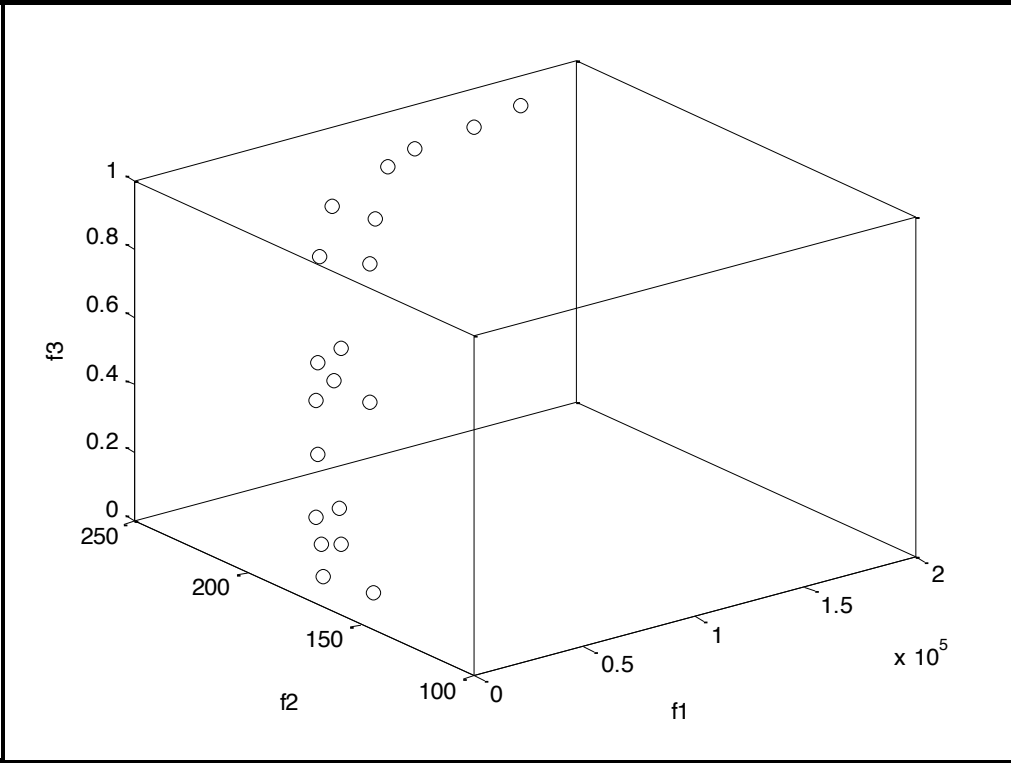
Gestion multi-objectif du réservoir

Solutions Pareto-optimales générées (200)

f1 - déficit d'irrigation
f2 – hydroélectricité
f3 – qualité de l'eau

Technique de partitionnement k-means pour réduire ce grand ensemble à des solutions Pareto-optimales représentatives





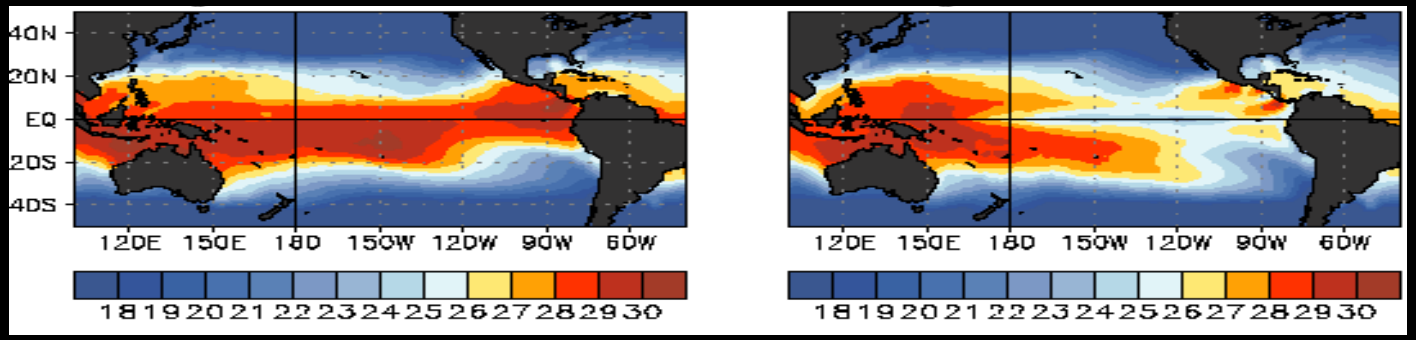


Prévision des apports aux réservoirs par réseaux de neurones face à la variabilité climatique

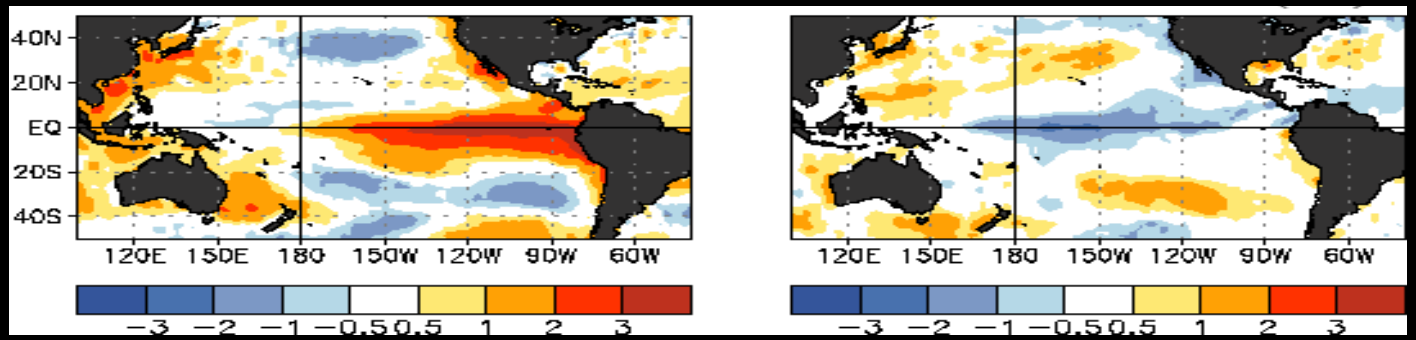
El Niño-Oscillation australe (ENSO)

Situation El Niño

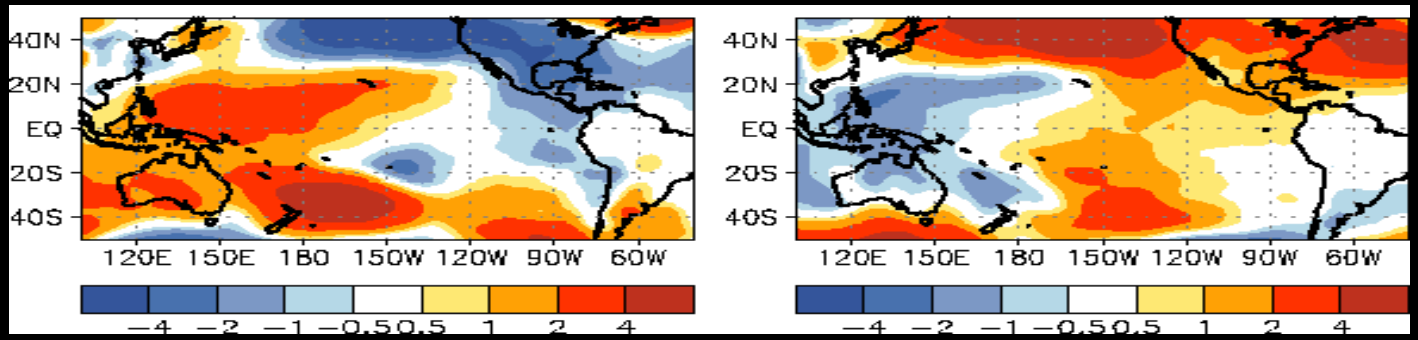
Situation La Niña



TSM (°C)



Anomalie de TSM (°C)



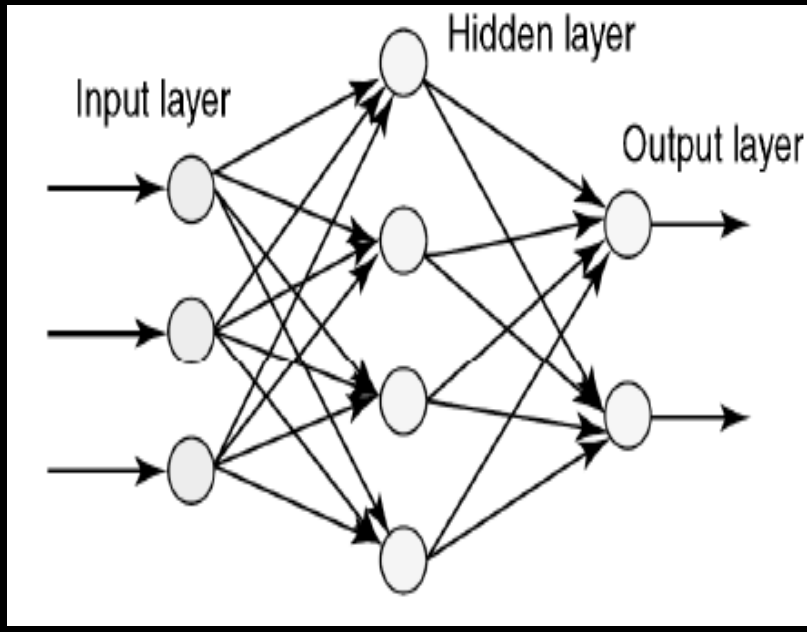
Anomalie de pression (mb)

Réseaux de neurones artificiels (RNA)

Caractéristiques des RNA standard :

-Apprentissage supervisé avec entraînement par lots

-Algorithme de rétropropagation : technique de descente de gradient qui minimise la fonction d'erreur du réseau.



RNA multicouches à propagation avant

-Fonction d'activation : Tangente sigmoïde

$$f(net) = \frac{2}{1+e^{-2(net)}} - 1$$

-Mesure de performance : MSE

$$E = \frac{1}{p\,q}\sum_{i=1}^{p}\sum_{j=1}^{q}(y_{i,j} - y_{i,j}^{t})^2$$

Réseaux de neurones PSO (RN-PSO)

Ici, les variables de décision sont les poids et les biais du réseau

Étape 1. Initialiser les paramètres
Étape 2. Générer la population initiale
Étape 3. Évaluer la fonction d'adaptation
Étape 4. Appliquer les opérations EMPSO aux variables de décision
Étape 4. Vérifier les critères d'arrêt

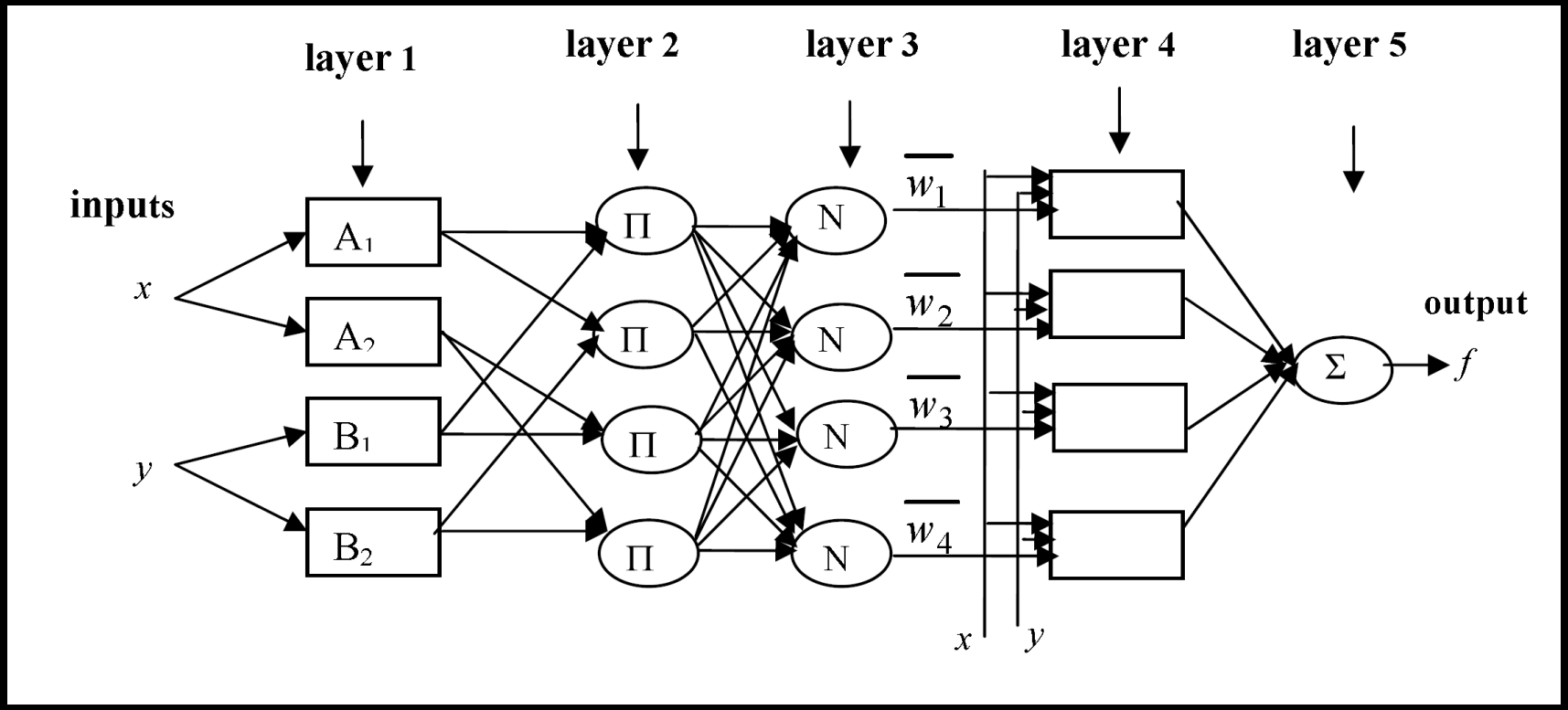
-Fonction d'activation :
Tangente sigmoïde

$$f(net) = \frac{2}{1 + e^{-2(net)}} - 1$$

-Mesure de performance :
MSE

$$E = \frac{1}{p\,q} \sum_{i=1}^{p} \sum_{j=1}^{q} (y_{i,j} - y_{i,j}^{t})^2$$

Système d'inférence floue à réseau adaptatif (ANFIS)



Couche 1 : prémisses (nœuds d'entrée)

Couche 2 : implication (nœuds de règles)

Couche 3 : normalisation (nœuds de moyenne)

Couche 4 : conséquents (nœuds conséquents)

Couche 5 : sortie (nœuds de sortie)

ANFIS

Couche 1 : prémisses (nœuds d'entrée)

$O_i^1 = \mu_{A_i}(x)$ pour i =1,2

$O_i^1 = \mu_{B_{i-2}}(y)$ pour i =3,4

$$\mu_{A_i} = \frac{1}{1 + \left| \frac{x - c_i}{a_i} \right|^{2b_i}}$$

Couche 2 : implication (nœuds de règles)

$$O_k^2 = w_k = \mu_{A_i}(x) * \mu_{B_j}(y)$$

Couche 3 : normalisation (nœuds de moyenne)

$$O_i^3 = \overline{w_i} = \frac{w_i}{\sum_{k=1}^{4} w_k}$$

Couche 4 : conséquents (nœuds conséquents)

$$O_i^4 = \overline{w_i}\,(p_i\,x + q_i\,y + r_i)$$

Couche 5 : sortie (nœuds de sortie)

$$O_i^5 = \overline{w_i}\,f_i = \frac{\sum_{i=1}^{4} w_i f_i}{\sum_{i=1}^{4} w_i}$$

Entraînement de l'ANFIS

•Dans la première étape, lors de la passe avant, la méthode des moindres carrés identifie les paramètres conséquents, tandis que les paramètres antécédents (paramètres des fonctions d'appartenance) sont supposés fixes pour le cycle en cours sur le jeu d'entraînement

•L'erreur entre les valeurs observées et le modèle entraîné est calculée, puis les signaux d'erreur sont rétropropagés. La descente de gradient met à jour et affine les paramètres des prémisses en minimisant la fonction de coût quadratique globale, tandis que les paramètres conséquents restent fixes

Étude de cas

Prévision des apports au réservoir de Malaprabha





Modèles de prévision des apports

Modèle de prévision des apports annuels

$$Q_t = f(Nino_{feb}, Nino_{mar}, Q_{t-2}, Q_{t-1})$$

Modèle de prévision des apports mensuels

$$Q_t = f(Nino_{t-2}, Nino_{t-1}, Q_{t-2}, Q_{t-1})$$

Modèle de prévision des apports décadaires

$$Q_t = f(Rain_{t-1}, Q_{t-2}, Q_{t-1})$$

Critères des modèles

- Périodicité
- Prétraitement des données
- Critères d'arrêt

Mesures de performance

Racine de l'erreur quadratique moyenne

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(Q_o^t - Q_m^t)^2}$$

Erreur absolue moyenne

$$MAE = \frac{1}{T}\sum_{t=1}^{T}\left|Q_o^t - Q_m^t\right|$$

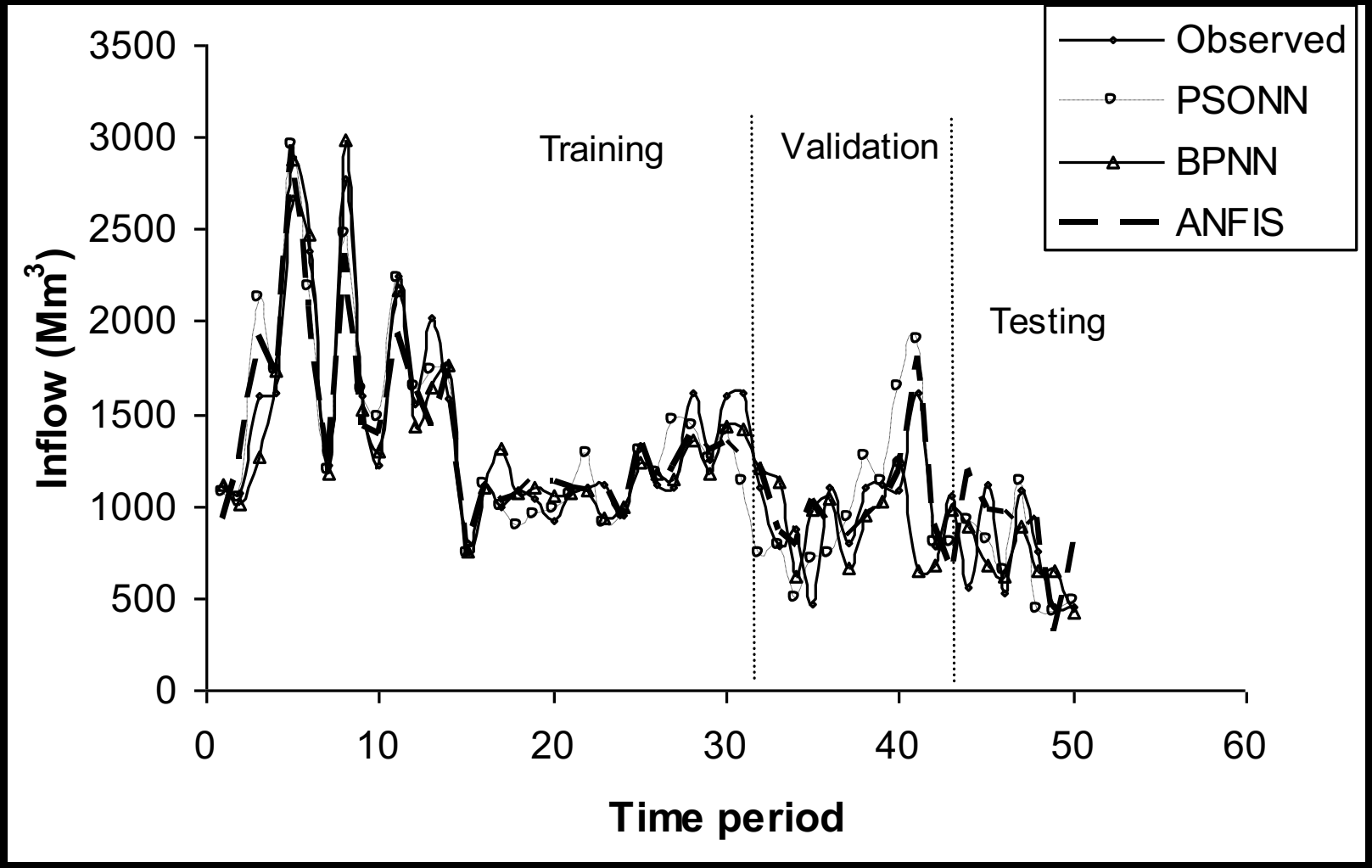
Coefficient de corrélation

$$r = \frac{\sum_{t=1}^{T}(Q_o^t - \overline{Q_o})(Q_m^t - \overline{Q_m})}{\sqrt{\sum_{t=1}^{T}(Q_o^t - \overline{Q_o})^2}\sqrt{\sum_{t=1}^{T}(Q_m^t - \overline{Q_m})^2}}$$

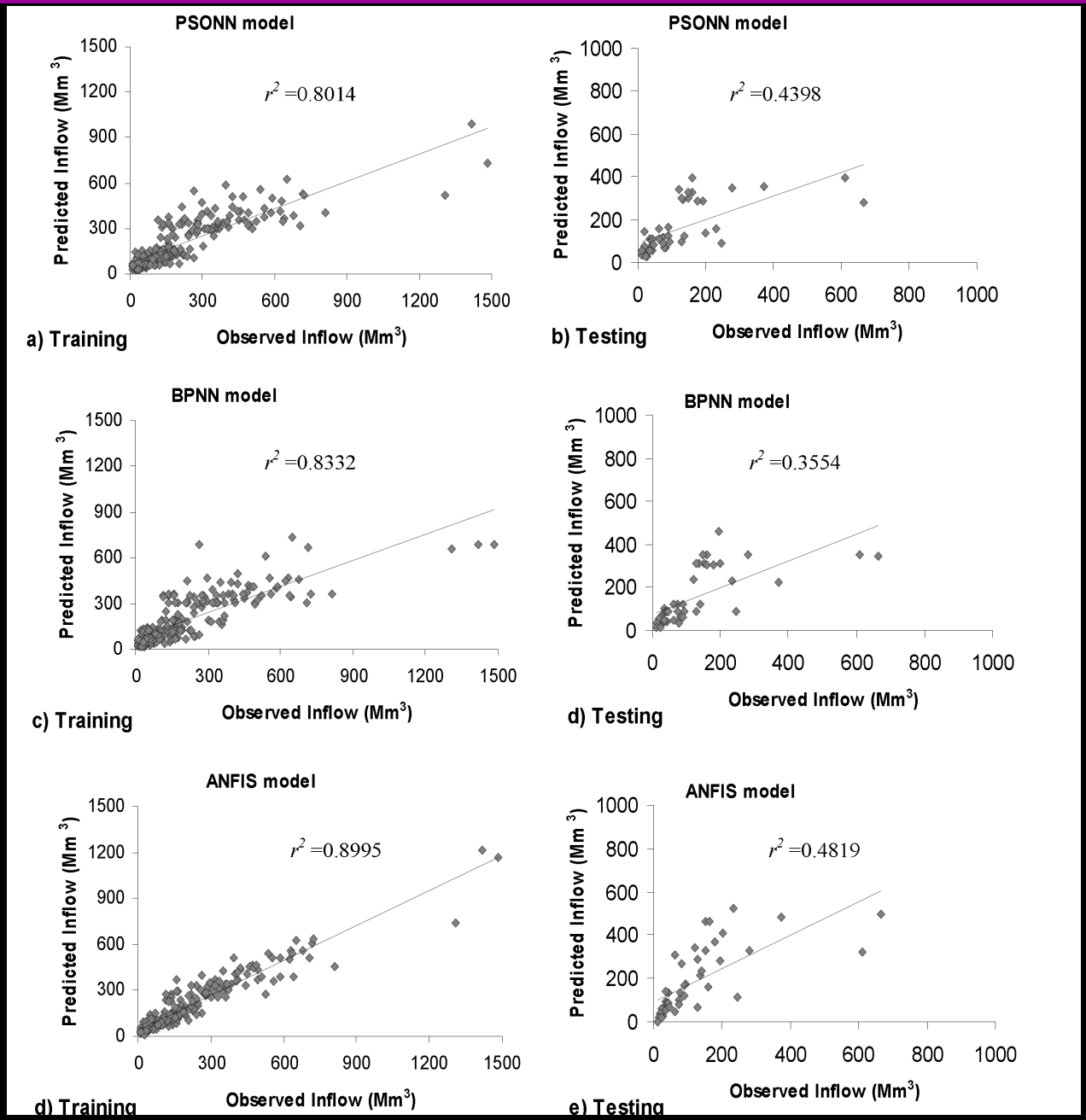
Coefficient d'efficacité de Nash-Sutcliffe

$$CE = 1 - \frac{\sum_{t=1}^{T}(Q_o^t - Q_m^t)^2}{\sum_{t=1}^{T}(Q_o^t - \overline{Q_o})^2}$$

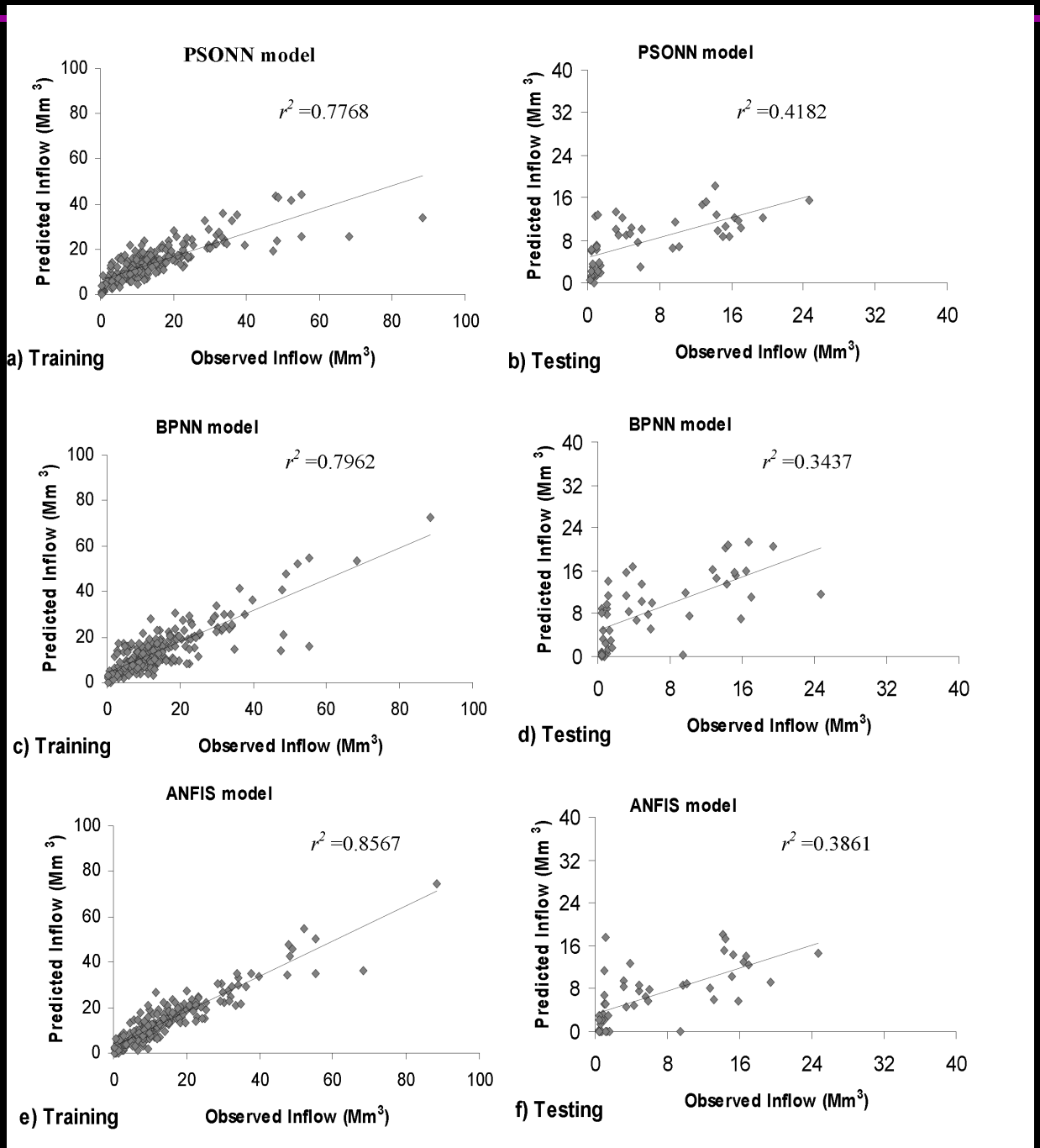
Résultats du modèle de prévision des apports annuels



Prévisions du modèle mensuel pour la saison de mousson



Prévisions du modèle mensuel pour la saison sèche



Statistiques de performance

Model		Performance measure	PSONN			BPNN			ANFIS		
			Training	Validation	Testing	Training	Validation	Testing	Training	Validation	Testing
Annual	*	*RMSE*	219.6361	309.127	213.7561	234.384	348.4827	292.4613	181.3062	285.7193	271.6191
		MAE	149.8344	268.4584	161.9773	152.4845	306.4618	266.8942	109.9852	232.5193	237.0673
		r	0.9227	0.7575	**0.6327**	0.9185	0.6583	0.5935	0.9516	0.7922	0.6082
		CE	0.8803	0.2165	0.3171	0.8720	0.3256	0.2183	0.9023	0.4325	0.2467
monthly	Monsoon season	*RMSE*	124.2218	88.2118	169.6542	89.5654	98.1284	201.8452	79.0521	74.3287	127.7455
		MAE	72.9947	62.2102	108.9456	64.5592	76.5942	129.9456	45.0099	48.407	90.3816
		r	0.8952	0.8582	0.6632	0.9128	0.65981	0.5962	0.9484	0.9023	**0.6942**
		CE	0.6924	0.7318	0.2313	0.6984	0.5784	0.3384	0.8754	0.8096	0.1834
	Non-monsoon season	*RMSE*	7.5010	6.6121	5.0251	6.8010	7.2121	7.864	4.8216	5.5417	6.9079
		MAE	4.5204	4.7637	3.5612	4.446	5.932	4.9318	3.0864	4.0361	4.1862
		r	0.8814	0.7419	**0.6467**	0.8923	0.5897	0.5863	0.9256	0.8278	0.6213
		CE	0.6299	0.5464	0.4221	0.7639	0.6954	0.32949	0.8471	0.6814	0.409
Ten daily	Monsoon season	*RMSE*	48.1687	55.6173	42.7136	56.5813	62.6837	47.5461	41.1305	53.0554	50.6004
		MAE	27.7492	28.154	24.7922	315873	29.856	26.5829	23.2053	25.0324	28.3243
		r	0.9106	0.7634	**0.6315**	0.8964	0.751	0.6048	0.9296	0.7781	0.5981
		CE	0.692	0.5579	0.3391	0.6234	0.5892	0.3251	0.6494	0.5977	0.3158
	Non-monsoon season	*RMSE*	2.4704	2.5779	2.0503	3.0782	3.8574	2.8569	2.1115	2.158	1.7649
		MAE	1.7180	1.6349	1.5807	2.3351	2.5694	1.6963	1.3777	1.4912	1.4354
		r	0.8915	0.7763	0.6152	0.8625	0.6945	0.5962	0.9126	0.7955	**0.6226**
		CE	0.5328	0.4455	0.2333	0.6482	0.6195	0.2259	0.6587	0.6114	0.4319

Gestion du réservoir en temps réel

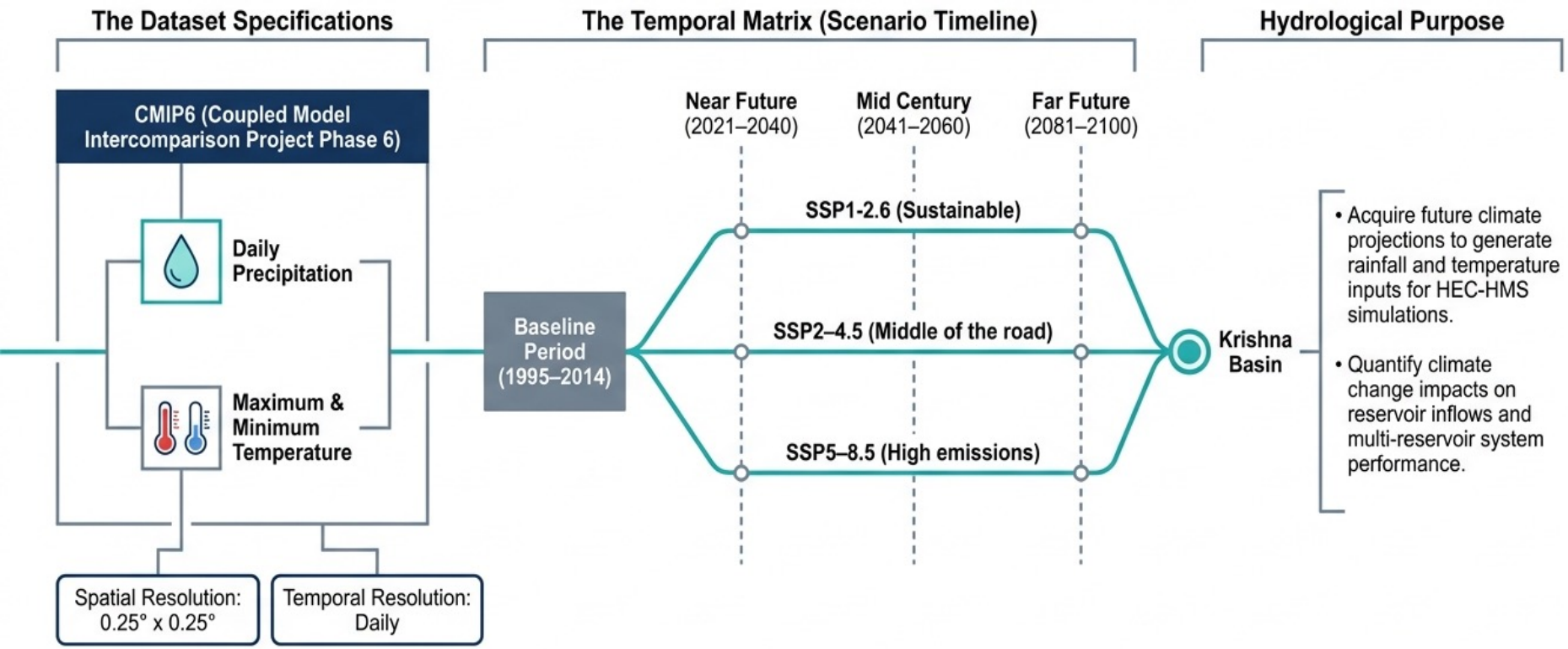
Prévoir les apports annuels en début de saison

Décider de la gestion optimale du réservoir avec le modèle multi-objectif

Mettre en œuvre la gestion du réservoir en temps réel ↔

- Prévisions décadaires des apports avec le modèle RN-PSO
- Modèle de gestion optimale du réservoir
- Technique EM-PSO pour l'optimisation

CMIP6 Climate Data Architecture and Scenario Selection

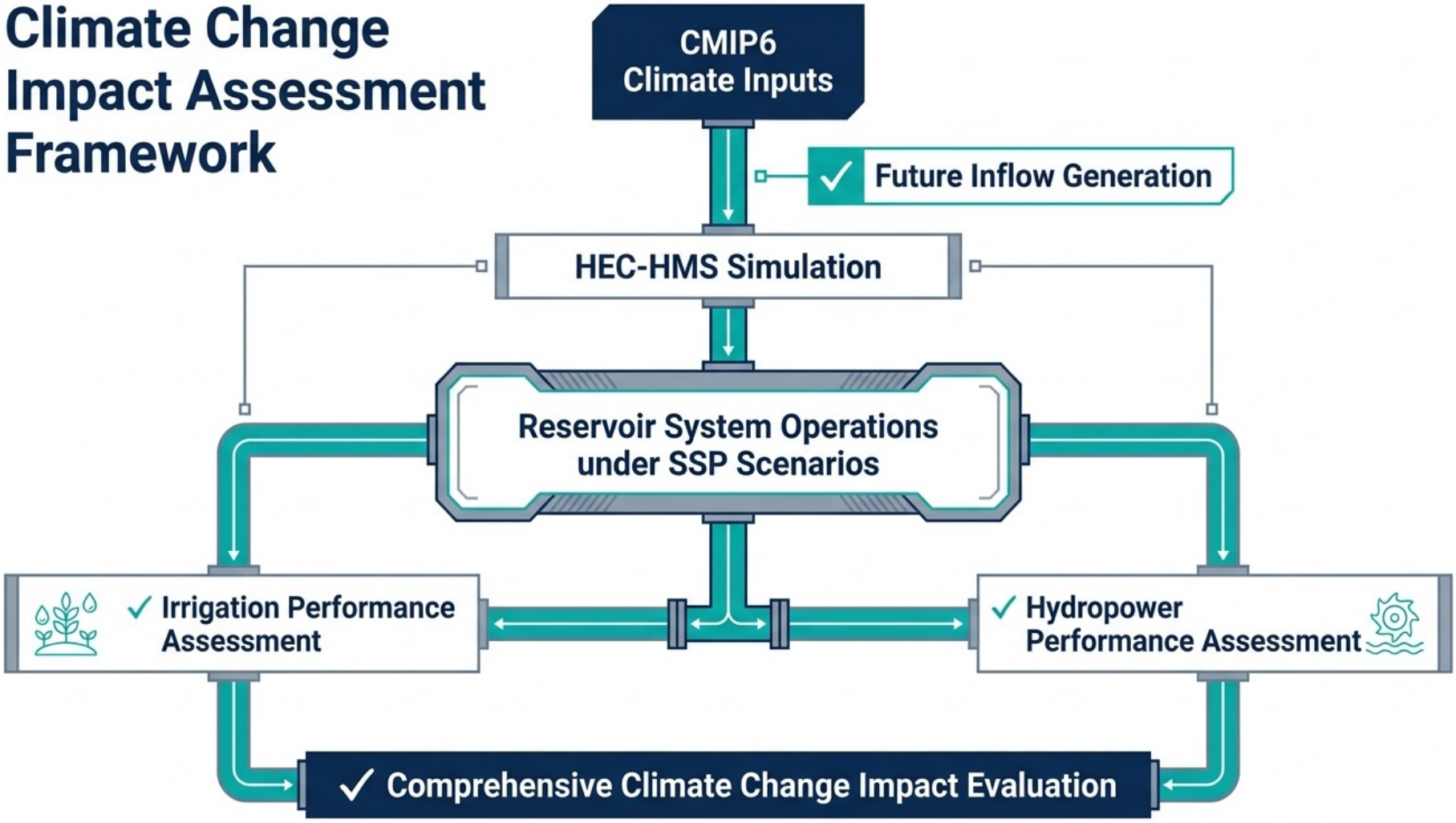


CMIP6 - Projet d'intercomparaison de modèles couplés, phase 6

SSP - trajectoires socio-économiques partagées

SSP 4.5 - SSP avec un forçage radiatif cible de 4,5 W/m²

Climate Change Impact Assessment Framework



Conclusions

- Les algorithmes génétiques sont de puissantes techniques d'optimisation non linéaire
- Ils peuvent servir à la gestion optimale multi-objectif d'un système de réservoirs
- Différentes formes de RNA jouent un rôle crucial dans la prévision des apports pour la gestion des réservoirs en temps réel

Perspectives

- CNN (1D et 2D)
- Réseaux Transformer
- L'IA pour la fouille de données
- PINN
- IA explicable

- Système d'aide à la décision pour une gestion optimale des réservoirs, intégrée et résiliente au climat, fondée sur l'IA

H2Open Journal

IWA PUBLISHING

© 2020 The Authors
135

H2Open Journal Vol 3 No 1
doi: 10.2166/h2oj.2020.128

Evolutionary algorithms, swarm intelligence methods, and their applications in water resources engineering: a state-of-the-art review

M. Janga Reddy [a,*] and D. Nagesh Kumar [b]

[a] Department of Civil Engineering, Indian Institute of Technology Bombay, Mumbai 400076, India

[b] Department of Civil Engineering, Indian Institute of Science, Bangalore 560012, India

Komaragiri Srinivasa Raju · Dasika Nagesh Kumar

Artificial Intelligence and Machine Learning Techniques in Engineering and Management

The present book covers various facets of Artificial Intelligence, Machine Learning, and Fuzzy Logic. It includes a brief discussion on performance indicators, Classical and Advanced Machine Learning algorithms, Fuzzy logic-based modelling algorithms, Emerging Research Areas, including Blockchain, recent ML techniques, Evolutionary Algorithms, Large Language Model (LLM)-based Generative AI, the Internet of Things, Big Data, Decision Support Systems, Taguchi design of experiments, data augmentation, and Cross-Validation, and representative case studies. The appendix covers representative AI tools, data sources, books, and journals on AI. The present book can support undergraduate, postgraduate, and Ph.D. students in Artificial Intelligence, Generative Artificial Intelligence, Machine Learning, Data Sciences, Soft Computing, and Fuzzy Logic in Engineering and Management and allied fields. The proposed book has immense value in the interdisciplinary and cross-disciplinary context.

ISBN 978-981-96-2620-5

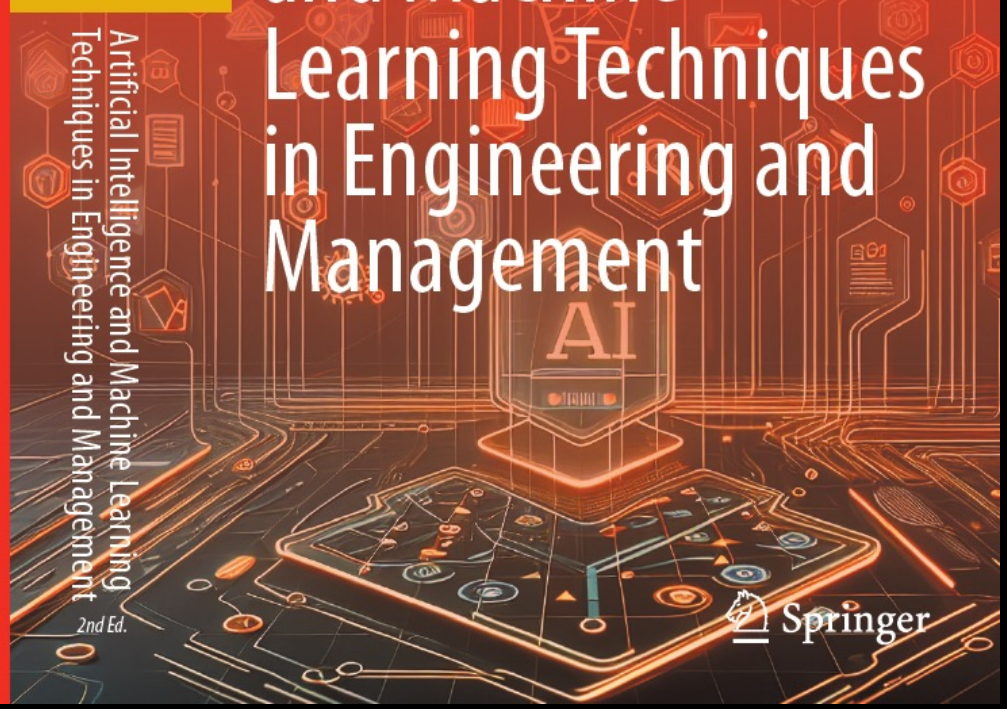


▶ springer.com

Srinivasa Raju · Nagesh Kumar

Komaragiri Srinivasa Raju
Dasika Nagesh Kumar

Artificial Intelligence and Machine Learning Techniques in Engineering and Management



Artificial Intelligence and Machine Learning Techniques in Engineering and Management 2nd Ed.

Springer



Merci

http://www.civil.iisc.ac.in/~nagesh