



UNIVERSITÉ DE GENÈVE

FACULTÉ DES SCIENCES

Département d'informatique

MASTER OF COMPUTER SCIENCE

MÉTAHEURISTIQUES POUR L'OPTIMISATION
HOMEWORK ASSIGNMENT REPORT

Series 6 : Genetic Algorithms and Function Minimization

Author :

Mohammad Oday DARWICH

Professors :

M. Bastien CHOPARD

Mme. Aziza MERZOUKI

M. Gregor CHLIAMOVITCH

21 novembre 2015

0.1 Introduction

Genetic algorithms have been used in science and engineering as adaptive algorithms for solving practical problems and as computational models of natural evolutionary systems.

But in more concrete term what is GA ?

A genetic algorithm (or GA) is a search technique used in computing to find true or approximate solutions to optimization and search problems.

- (GA)s are categorized as global search heuristics.
- (GA)s are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination)
- The evolution usually starts from a population of randomly generated individuals and happens in generations.
- In each generation, the fitness of every individual in the population is evaluated, multiple individuals are selected from the current population (based on their fitness), and modified to form a new population.
- The new population is used in the next iteration of the algorithm.[?]

generation=0 initialize population while (not end-condition) generation+=1 compute the fitness of each individual select individuals crossover mutation end while

0.2 Pseudo code of GA and how we implement it

Algorithm 1 GA Genetic algorithm

```

1: generation = 0
2: initialize population
3: while (not end-condition) do
4:   generation += 1
5:   compute the fitness of each individual
6:   select individuals
7:   crossover
8:   mutation
9: end while
10:  $P(t) \xrightarrow{\text{Select}} P'(t) \xrightarrow{\text{crossover}} P''(t) \xrightarrow{\text{mutation}} P'''(t) = P(t+1)$ 

```

So how it works the pseudo code above, matching with parameters and criteria of this TP :

After initializing the population of size 100, *pop_size*, we compute the *fitness function* from the following fitness function :

$$f(x, y) = -\left|\frac{1}{2} x \sin(\sqrt{|x|}) - |y \sin(30 \sqrt{|\frac{x}{y}|})\right| \quad (1)$$

where $x, y \in [10, 1000] \cap \mathbb{N}$

- **Step 1 : select individual**¹ consist of selecting an individual from the population, the one with the best cost and to do that here we use the *k-tournament* method, here we use the 5-tournament.
- **Step 2 : crossover**² we mate strings of two parents for crossover. For each couple we first decide (using some pre-defined probability $P_{crossover}$ ³, for instance 0.6) and here an example showing before and after crossover
 - ↪ **Before** $s'_1 = 1111010101$ $s'_2 = 1110110101$
 - ↪ **After** $s'_1 = 1110110101$ $s'_2 = 1111010101$
- **The final step : mutation**⁴ for these n new individuals we mutate with $P_{mutation} = 0.01$ ⁵ in the 1st experiment and $P_{mutation} = 0.1$ in the 2nd experiment, and the mutation is as the following :
 - ↪ **Before** $s''_1 = 1110110101$ $s''_2 = 1111010101$

1. We will see in the result that this operator tend to the intensification
 2. the crossover and mutation tend to the diversification
 3. $P_{crossover}$ in the code denoted *pc*
 4. The mutation is to make random perturbation in the coding of the individual
 5. $P_{mutation}$ in the code denoted *pm*

↻ After $s_1''' = 1110100101$ $s_2''' = 111110100$ and so on...

And now, we **iterate**, i a generation we can see the total population fitness change and we will see how it will be improved.

So at this point, we go through the same process all over again, **until the stopping criterion is met**.

0.3 Done Work

0.3.1 Function visualization

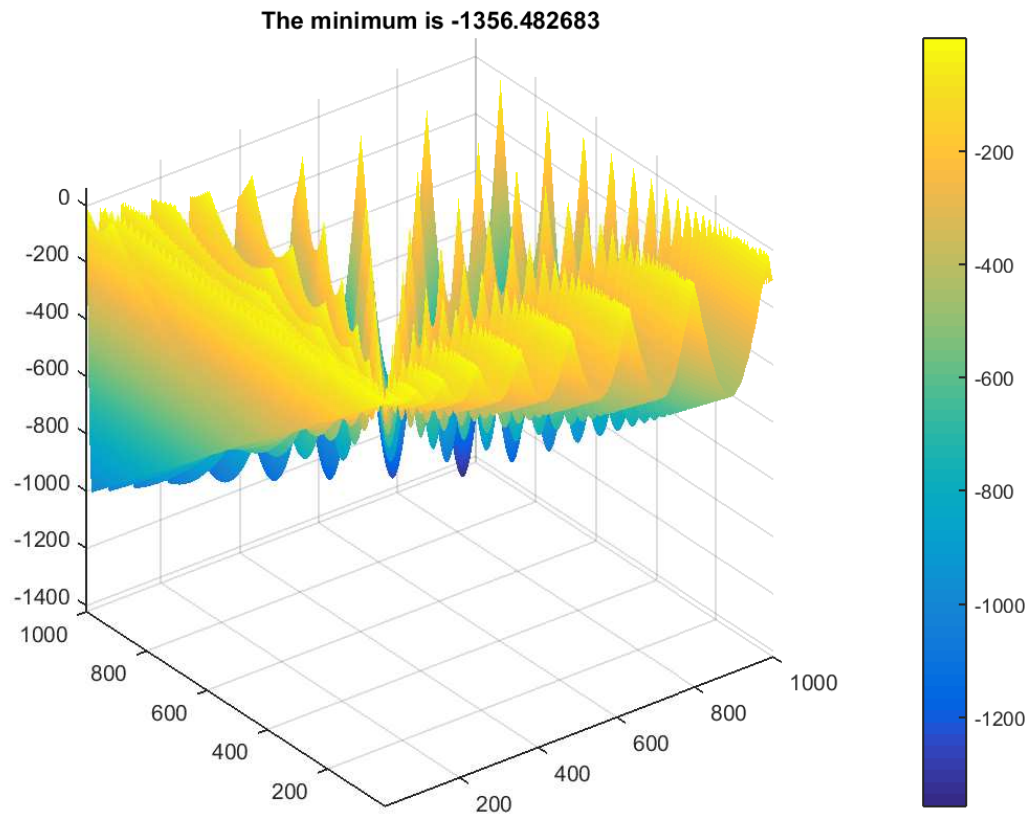


FIGURE 1 – Path journey

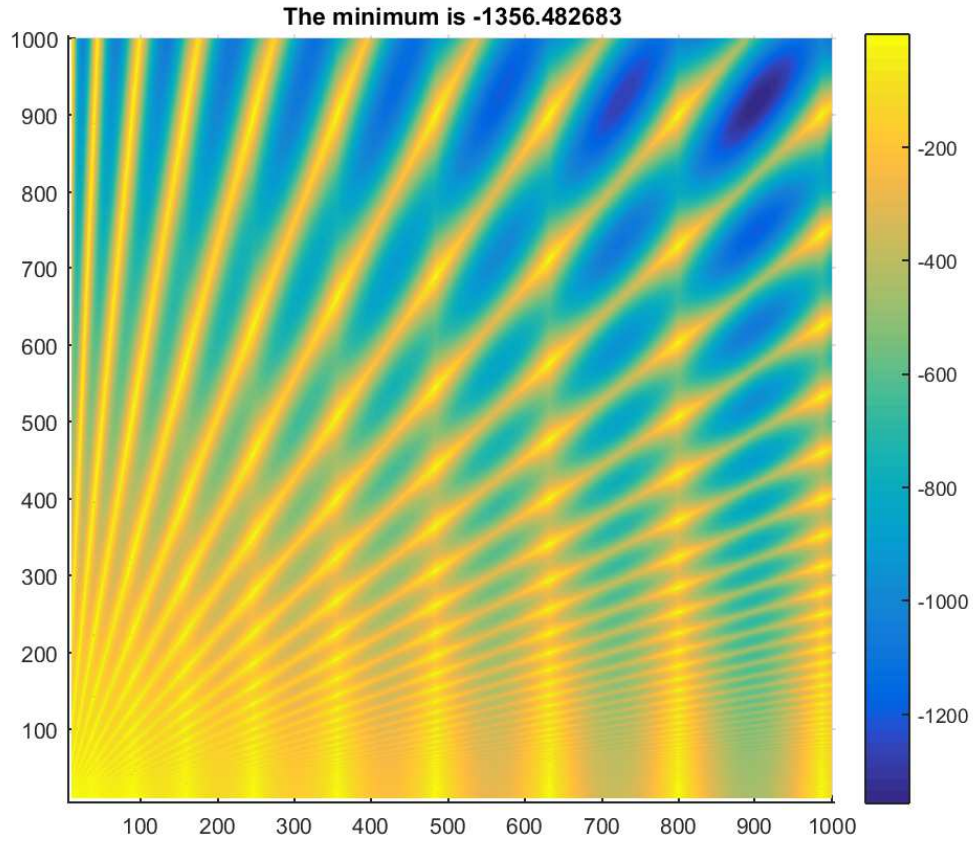


FIGURE 2 – 3D visualization

0.3.2 Tests and evaluation

Experiments with 2 differents probabilities of mutation pm and crossover pc :

In the following example we will see the different experiments following to this table :

TABLE 1 – Table of parameteres of done experiments

pc	0.0	0.0	0.6	0.6
pm	0.1	0.01	0.1	0.01
gmax	10^3	10^4	10^5	–

The selection gives more chance to particular individuals of good quality to be in the next population. But of course we have to add an element of innovation, without which no new solution can not emerge. The change required is provided by the genetic operators of crossover and mutation. And as you see below i did many experiments varying these parameters to see the differences.

We see below the results after 10^3 evaluation, with the mean of fitness and the STD and the success rate.

0.3.3 interpretation of results

- From the results below, we see that without crossover and with prob of mutation equal to 0.1 the success rate is 0.08 and it go down to zero with $P_m = 0.01$ and it the same case for 10^4 and 10^5 but the success rate go up respectively to 0.28 and 0.48 so more the number execution increase more the success rate increase.
- we see that in the space of generation the total is ameliorated for 2.28 % .

This result is also like that because of the non diversity (without crossover) so we see that the mutation has a negative effect even if occasionally it can ameliorate the results.

TABLE 2 – Exp1 for after 10^5 evaluation

Exp1 for after 10^5 evaluation			
	fit		fit
run nb :1	-1356.48	run nb :1	-1356.40
run nb :2	-1356.48	run nb :2	-1356.40
run nb :3	-1356.48	run nb :3	-1356.40
run nb :4	-1356.48	run nb :4	-1356.40
run nb :5	-1356.39	run nb :5	-1356.40
run nb :6	-1356.48	run nb :6	-1356.40
run nb :7	-1356.48	run nb :7	-1356.40
run nb :8	-1356.48	run nb :8	-1356.40
run nb :9	-1356.28	run nb :9	-1356.40
run nb :10	-1356.48	run nb :10	-1356.40
run nb :11	-1356.45	run nb :11	-1356.40
run nb :12	-1356.45	run nb :12	-1355.95
run nb :13	-1356.45	run nb :13	-1356.40
run nb :14	-1356.48	run nb :14	-1356.40
run nb :15	-1356.48	run nb :15	-1356.40
run nb :16	-1356.48	run nb :16	-1356.40
run nb :17	-1356.48	run nb :17	-1356.40
run nb :18	-1356.44	run nb :18	-1356.40
run nb :19	-1356.29	run nb :19	-1356.40
run nb :20	-1356.40	run nb :20	-1356.40
run nb :21	-1356.29	run nb :21	-1356.40
run nb :22	-1356.48	run nb :22	-1356.40
run nb :23	-1356.40	run nb :23	-1356.40
run nb :24	-1356.48	run nb :24	-1356.40
run nb :25	-1356.48	run nb :25	-1356.40
STD	0.07	STD	0.09
Mean	-1356.44	Mean	-1356.38
P_c	0.60	P_c	0.60
P_m	0.10	P_m	0.01
Succrate	0.56	Succrate	0.00
generation	1000.00	generation	1000.00

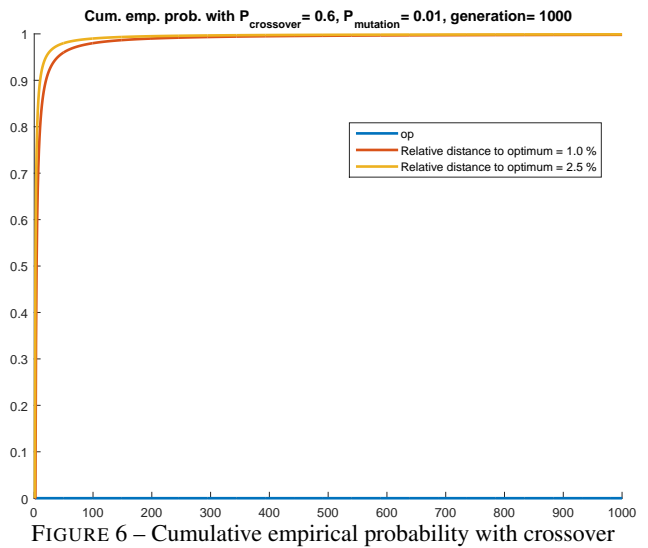
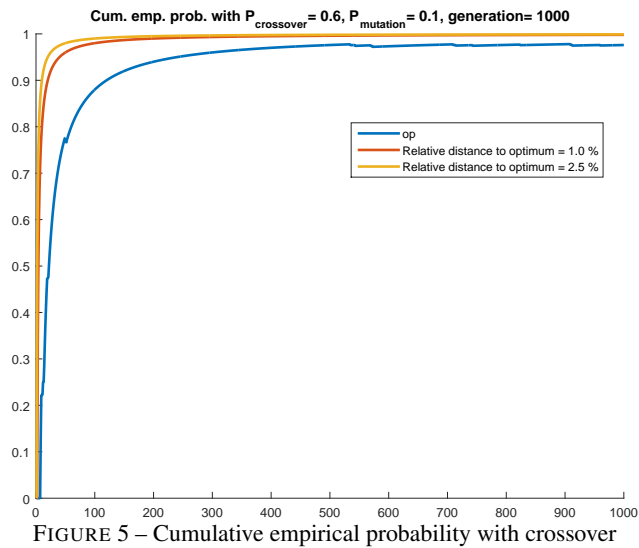
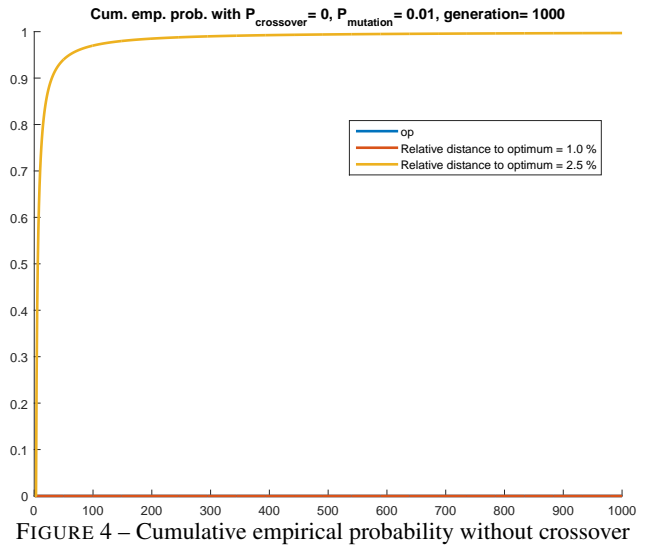
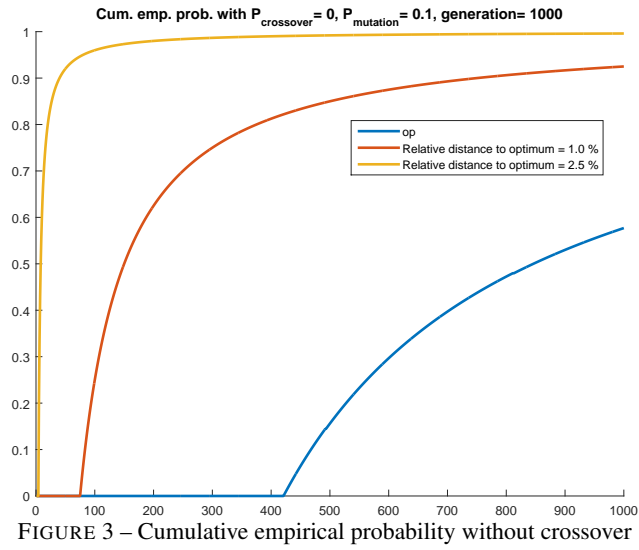
	fit
run nb :1	-1350.59
run nb :2	-1356.40
run nb :3	-1350.59
run nb :4	-1356.29
run nb :5	-1350.59
run nb :6	-1350.59
run nb :7	-1350.59
run nb :8	-1350.59
run nb :9	-1350.59
run nb :10	-1350.59
run nb :11	-1350.59
run nb :12	-1350.59
run nb :13	-1350.59
run nb :14	-1350.59
run nb :15	-1350.59
run nb :16	-1350.59
run nb :17	-1350.59
run nb :18	-1350.59
run nb :19	-1350.59
run nb :20	-1350.59
run nb :21	-1350.59
run nb :22	-1350.59
run nb :23	-1350.59
run nb :24	-1350.59
run nb :25	-1350.59
STD	1.59
Mean	-1351.05
P_c	0.00
P_m	0.01
Succrate	0.00
generation	1000.00

The average number of iteration to reach the optimum value for all the experiments in the table 1 it's 9, 20, 13, 9 with respectively and STD of 17, 30, 14, 3.

0.4 Discussion 1

After referring to the results on table 3, 4 and 2 we see that the mutation has an important impact on the quality of solution. and also we see that with a probability of crossover equal to 0.6 we have better results. so we have better result with an important percentage of probability of mutation and also for the mid-break crossover there are no improvement for the solutions, because in the function we have many local minima. However the mutation improve the solutions. so the mutation at the end avoid the local minima.

0.5 The cumulative empirical probability results and Discussion 2



From the figure 6 above we see that without crossover that the optimum in the optimum has not been reached, and this show that we have a local optimum, and that affirm what i just said before that mutation has a negative effect (more local minima).

The better result we have is when $P_m = 0.6$ and $P_c = 0.1$ as seen in the figure 5, best solution (fitness improved) with lower standard deviation. also the quality of solution do not vary so much as seen in the results of the tables on the annex.

0.6 Conclusion

As conclusion we can say that more value of generation increase more the fitness getting better. and at a number of generation like 9 in our case the optimal solution is found. but also the space of research was not blocked in all case. so this landscape fitness is easy for an algorithm AG.

The average fitness continues to improve over time showing a population becoming more and more homogeneous. And Because of the stochastic nature of evolutionary algorithms, their performance is not the same on various execution, unless one uses the same parameters and the same generation value. but in another side the average of many execution indicate the viability of the algorithm

Annex

TABLE 3 – Exp1 for after 10^3 evaluation

Exp1 for after 10^3 evaluation															
	fit			fit			fit			fit			fit		
run nb :1	-1356.45			run nb :1	-1356.48			run nb :1	-1356.05				run nb :1	-1281.95	
run nb :2	-1356.48			run nb :2	-1356.48			run nb :2	-1348.69				run nb :2	-1279.09	
run nb :3	-1356.44			run nb :3	-1354.52			run nb :3	-1351.65				run nb :3	-1279.09	
run nb :4	-1356.18			run nb :4	-1354.88			run nb :4	-1356.19				run nb :4	-1299.93	
run nb :5	-1356.39			run nb :5	-1355.36			run nb :5	-1352.47				run nb :5	-1279.09	
run nb :6	-1356.18			run nb :6	-1355.39			run nb :6	-1356.45				run nb :6	-1282.35	
run nb :7	-1356.03			run nb :7	-1351.24			run nb :7	-1356.20				run nb :7	-1280.19	
run nb :8	-1356.03			run nb :8	-1351.24			run nb :8	-1356.48				run nb :8	-1282.35	
run nb :9	-1356.29			run nb :9	-1355.95			run nb :9	-1356.48				run nb :9	-1280.19	
run nb :10	-1355.92			run nb :10	-1356.14			run nb :10	-1355.77				run nb :10	-1280.19	
run nb :11	-1356.05			run nb :11	-1351.24			run nb :11	-1356.19				run nb :11	-1280.19	
run nb :12	-1355.77			run nb :12	-1351.24			run nb :12	-1352.80				run nb :12	-1280.19	
run nb :13	-1356.48			run nb :13	-1351.24			run nb :13	-1356.48				run nb :13	-1280.19	
run nb :14	-1356.29			run nb :14	-1356.03			run nb :14	-1355.39				run nb :14	-1279.09	
run nb :15	-1350.63			run nb :15	-1356.48			run nb :15	-1355.75				run nb :15	-1277.08	
run nb :16	-1356.40			run nb :16	-1351.24			run nb :16	-1356.48				run nb :16	-1279.09	
run nb :17	-1356.45			run nb :17	-1354.52			run nb :17	-1356.40				run nb :17	-1279.09	
run nb :18	-1356.28			run nb :18	-1356.03			run nb :18	-1352.69				run nb :18	-1282.35	
run nb :19	-1355.32			run nb :19	-1356.48			run nb :19	-1355.29				run nb :19	-1318.98	
run nb :20	-1356.48			run nb :20	-1356.48			run nb :20	-1355.39				run nb :20	-1280.19	
run nb :21	-1356.29			run nb :21	-1351.24			run nb :21	-1349.83				run nb :21	-1281.95	
run nb :22	-1356.03			run nb :22	-1356.48			run nb :22	-1356.48				run nb :22	-1282.35	
run nb :23	-1355.77			run nb :23	-1351.24			run nb :23	-1352.39				run nb :23	-1280.19	
run nb :24	-1356.40			run nb :24	-1351.24			run nb :24	-1352.69				run nb :24	-1279.09	
run nb :25	-1356.48			run nb :25	-1351.24			run nb :25	-1352.47				run nb :25	-1282.35	
STD	1.15			STD	2.37			STD	2.32				STD	8.63	
Mean	-1355.98			Mean	-1354.01			Mean	-1354.53				Mean	-1282.67	
P_c	0.60			P_c	0.60			P_c	0.00				P_c	0.00	
P_m	0.10			P_m	0.01			P_m	0.10				P_m	0.01	
Succrate	0.08			Succrate	0.16			Succrate	0.08				Succrate	0.00	
generation	10.00			generation	10.00			generation	10.00				generation	10.00	

TABLE 4 – Exp1 for after 10^4 evaluation

Exp1 for after 10^4 evaluation															
	fit			fit			fit			fit			fit		
run nb :1	-1356.48			run nb :1	-1356.29			run nb :1	-1356.48				run nb :1	-1350.99	
run nb :2	-1356.48			run nb :2	-1356.48			run nb :2	-1356.48				run nb :2	-1350.99	
run nb :3	-1356.48			run nb :3	-1355.36			run nb :3	-1356.45				run nb :3	-1266.37	
run nb :4	-1352.69			run nb :4	-1355.36			run nb :4	-1356.48				run nb :4	-1350.99	
run nb :5	-1356.48			run nb :5	-1356.48			run nb :5	-1356.48				run nb :5	-1266.68	
run nb :6	-1356.48			run nb :6	-1356.29			run nb :6	-1356.48				run nb :6	-1350.99	
run nb :7	-1356.48			run nb :7	-1317.43			run nb :7	-1338.00				run nb :7	-1350.99	
run nb :8	-1356.20			run nb :8	-1356.40			run nb :8	-1356.48				run nb :8	-1271.45	
run nb :9	-1338.00			run nb :9	-1317.43			run nb :9	-1356.45				run nb :9	-1317.43	
run nb :10	-1356.39			run nb :10	-1355.36			run nb :10	-1356.45				run nb :10	-1350.59	
run nb :11	-1356.48			run nb :11	-1356.45			run nb :11	-1352.47				run nb :11	-1355.36	
run nb :12	-1356.40			run nb :12	-1317.43			run nb :12	-1356.48				run nb :12	-1350.99	
run nb :13	-1356.48			run nb :13	-1355.75			run nb :13	-1356.19				run nb :13	-1266.68	
run nb :14	-1356.48			run nb :14	-1356.48			run nb :14	-1356.48				run nb :14	-1348.99	
run nb :15	-1352.69			run nb :15	-1317.43			run nb :15	-1356.48				run nb :15	-1350.99	
run nb :16	-1352.69			run nb :16	-1356.29			run nb :16	-1356.48				run nb :16	-1350.99	
run nb :17	-1337.91			run nb :17	-1356.45			run nb :17	-1356.48				run nb :17	-1356.40	
run nb :18	-1352.69			run nb :18	-1355.29			run nb :18	-1356.39				run nb :18	-1350.59	
run nb :19	-1352.66			run nb :19	-1356.29			run nb :19	-1356.44				run nb :19	-1350.99	
run nb :20	-1352.66			run nb :20	-1356.45			run nb :20	-1356.48				run nb :20	-1350.99	
run nb :21	-1356.48			run nb :21	-1317.43			run nb :21	-1356.40				run nb :21	-1350.99	
run nb :22	-1356.40			run nb :22	-1356.45			run nb :22	-1352.66				run nb :22	-1356.40	
run nb :23	-1356.48			run nb :23	-1355.36			run nb :23	-1356.48				run nb :23	-1350.99	
run nb :24	-1337.91			run nb :24	-1317.43			run nb :24	-1356.48				run nb :24	-1350.59	
run nb :25	-1356.48			run nb :25	-1317.43			run nb :25	-1356.19				run nb :25	-1356.40	
STD	6.02			STD	17.71			STD	3.78				STD	31.67	
Mean	-1353.32			Mean	-1345.25			Mean	-1355.39				Mean	-1337.03	
P_c	0.60			P_c	0.60			P_c	0.00				P_c	0.00	
P_m	0.10			P_m	0.01			P_m	0.10				P_m	0.01	
Succrate	0.16			Succrate	0.04			Succrate	0.28				Succrate	0.00	
generation	100.00			generation	100.00			generation	100.00				generation	100.00	

0.7 Elucider l'algorithme

0.7.1 Propriétés de l'algorithme

Une méthaheuristique de colonie de fourmis est un processus stochastique construisant une solution, en ajoutant des composants aux solutions partielles. Ce processus prend en compte

- une heuristique du problème
- des pistes de phéromone changeant dynamiquement pour refléter l'expérience acquise par les agents.

Dans l'algorithme AS à chaque itération t ($1 \leq t \leq t_{max}$), chaque fourmi K ($k = 1, \dots, m$) parcourt le graphe et construit un trajet complet de $n = |N|$ étapes. Pour chaque fourmi, le trajet entre une ville i et une ville j dépend de :

1. La liste des villes déjà visitées, qui définit les mouvements possibles à chaque pas, quand la fourmi k est sur la ville i : J_i^k ;
2. l'inverse de la distance entre les villes : $\eta_{ij} = \frac{1}{d_{ij}}$, *the heuristic information* cette information statique est utilisée pour diriger le choix des fourmis vers des villes proches, et éviter les villes trop lointaines ;
3. la quantité de phéromone déposée sur l'arête reliant les deux villes. Ce paramètre définit l'attractivité d'une partie du trajet global et change à chaque passage d'une fourmi. C'est en quelque sorte une mémoire globale du système, qui évolue par apprentissage.

La règle de déplacement est alors la suivante :

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}(t)^\alpha (\eta_{ij})^\beta}{\sum_{\ell \in J_i^k} \tau_{i\ell}(t)^\alpha (\eta_{i\ell})^\beta} & \text{if } j \in J_i^k \\ 0 & \text{otherwise } \sim j \notin J_i^k \end{cases} \quad \text{où} \quad (2)$$

— α et β contrôlent l'importance relative du phéromone versus *the heuristic information* η_{ij}

Après un tour complet (chaque itération), chaque fourmi laisse une certaine quantité de phéromones $\Delta\tau_{ij}^k(t)$ sur l'ensemble de son parcours, quantité qui dépend de la *qualité* de la solution trouvée :

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{L^k(t)} & \text{if ant } k \text{ used edge } (i,j) \text{ in its tour} \\ 0 & \text{otherwise} \end{cases} \quad \begin{matrix} \sim \text{if } (i,j) \in T_k(t) \\ \sim \text{if } (i,j) \notin T_k(t) \end{matrix} \quad \text{où} \quad (3)$$

— $T_k(t)$ est le trajet effectué par la fourmi k à l'itération t ; $L^k(t)$ est la longueur du tour ; Q is a constant ;

Comme on a dans l'énoncé l'équation de mise à jour du phéromone τ_{ij} donnée par :

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad (4)$$

Où

- m est le nombre de fourmis ;
- ρ est l'évaporation du phéromones ;

En effet pour éviter d'être piégé dans des solutions sous-optimales, il est nécessaire de permettre au système d'oublier les mauvaises solutions. On contrebalance donc l'additivité des phéromones par une décroissance constante des valeurs des arêtes à chaque itération.

0.8 Interpretation sur α and β Intensification et diversification

Avec $\alpha = 0$, seul la *heuristic information* est prise en compte ; la ville la plus proche est donc choisie à chaque pas. Au contraire, avec $\beta = 0$, seules l'intensité de phéromone jouent. on peut voir ci dessous les résultat si on donne à $\alpha = 0$ et $\beta = 5$

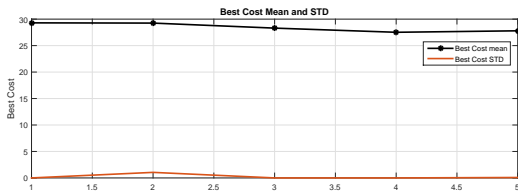


FIGURE 7 – Mean and STD variation of Best Cost $\alpha = 0$

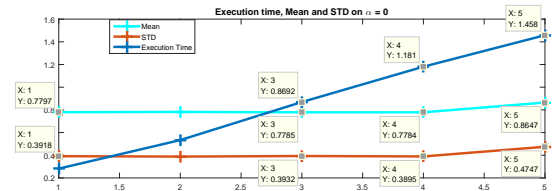


FIGURE 8 – Execution time, Mean and STD on $\alpha = 0$

Alors pour éviter une sélection trop rapide d'un trajet, un compromis entre ces deux paramètres, jouant sur les comportements de *diversification* et d'*intensification* est nécessaire.

α et β déterminent l'influence relative des pistes de phéromone et de l'information heuristique.

plus la valeur de α sera élevé plus l'*intensification* sera importante, car plus les pistes auront une influence sur le choix des fourmis. À l'inverse, plus α sera faible, plus la *diversification* sera forte car les fourmis éviteront les pistes. β agit de façon laire. On doit donc gérer à la fois les deux paramètres pour régler ces aspects. et ci dessous les résultats confirme ce qu'on dit

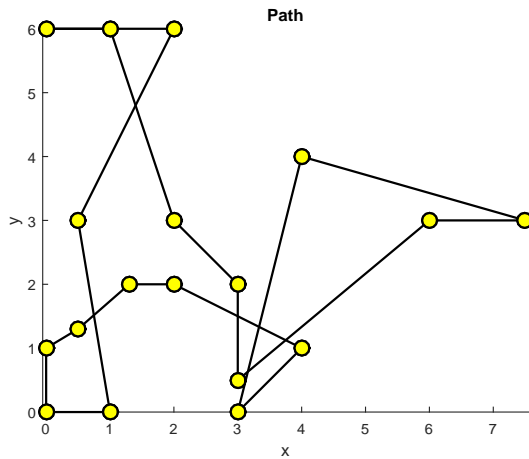


FIGURE 9 – Path avec diminution de la valeur de $\beta = 1$
On peut bien remarque l'influence du changement de la valeur de β quand par comparaison avec $\beta = 5$.

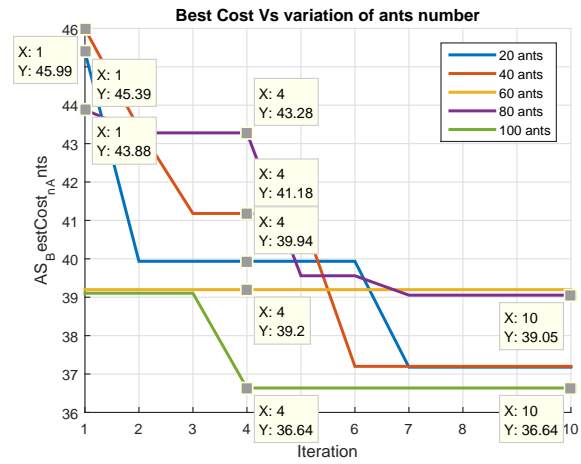


FIGURE 10 – Best Cost with ants variation $\beta = 1$ $\alpha = 0$
On peut bien remarque l'influence du changement de la valeur de β quand par comparaison avec $\beta = 5$.