

Higher National School of Hydraulic

The Library

Digital Repository of ENSH



المدرسة الوطنية العليا للري

المكتبة

المستودع الرقمي للمدرسة العليا للري



The title (العنوان):

Contribution à l'optimisation d'un système d'AEP par utilisation du logiciel Epanet et des algorithmes génétiques - Application sur le réseau du Sidi Lakhdar - (W. Ain defla).

The paper document Shelf mark (الشفرة) : 6-0016-18

APA Citation (توثيق APA):

Drici, Karima (2018). Contribution à l'optimisation d'un système d'AEP par utilisation du logiciel Epanet et des algorithmes génétiques - Application sur le réseau du Sidi Lakhdar - (W. Ain defla)[Thèse de master, ENSH].

The digital repository of the Higher National School for Hydraulics "Digital Repository of ENSH" is a platform for valuing the scientific production of the school's teachers and researchers.

Digital Repository of ENSH aims to limit scientific production, whether published or unpublished (theses, pedagogical publications, periodical articles, books...) and broadcasting it online.

Digital Repository of ENSH is built on the open DSpace software platform and is managed by the Library of the National Higher School for Hydraulics. <http://dspace.ensh.dz/jspui/>

المستودع الرقمي للمدرسة الوطنية العليا للري هو منصة خاصة بتقييم الإنتاج العلمي لأساتذة و باحثي المدرسة.

يهدف المستودع الرقمي للمدرسة إلى حصر الإنتاج العلمي سواء كان منشورا أو غير منشور (أطروحات، مطبوعات، مبداعات، مقالات، البوريات، كتب....) و بنه على الخط.

المستودع الرقمي للمدرسة مبني على المنصة المفتوحة DSpace و يتم إدارته من طرف مديرية المكتبة للمدرسة العليا للري.

كل الحقوق محفوظة للمدرسة الوطنية العليا للري.



MEMOIRE DE MASTER

Pour l'obtention du diplôme de Master en Hydraulique

Option: ALIMENTATION EN EAU POTABLE

THEME :

Contribution à l'optimisation d'un système d'AEP par utilisation du logiciel Epanet et des algorithmes génétiques - Application sur le réseau du SIDI LAKHDAR - (W.Ain defla)

Présenté par :

DRICI Karima

Devant les membres du jury

Nom et Prénoms	Grade	Qualité
SALAH Boualem	Professeur	Président
HACHEMI Abdelkader	M.C.B	Examineur
MOKRANE Wahiba	M.A.A	Examineur
TAFAT Leila	M.A.A	Examineur
DERNOUNI Youcef	M.A.A	Promoteur

Session Septembre 2019

Remerciements

Avant toute chose nous devons remercier Allah le tout puissant pour la volonté, le courage et la patience qu'il nous a apporté pour pouvoir réaliser ce travail.

Toute ma gratitude pour Monsieur D.ZEKRIFA pour ses judicieux conseils ainsi je remercie mon promoteur Mr DERNOUNI sur sa disponibilité tout long de l'élaboration et ses conseils.

A Monsieur le professeur B.SALAH. qui me fait l'honneur de présider notre jury. Je le remercie beaucoup parce qu'il m'a guidé plusieurs fois pour compléter mon travail.

A Monsieur A.HACHEMI , Madame W.MOKRANE et Madame L.TAFAT. qui m'ont fait l'honneur d'examiner mon travail. Je vous remercie également de nous transmettre votre savoir au cours de notre cursus universitaire.




En fin, nous adressons nos sincères remerciements et nos profonds respects à nos très chères familles qui ont été toujours là avec nous, et qui, grâce à eux, nous sommes là aujourd'hui, à nos amis et à toute personne qui a contribué même avec un mot de soutien.

Nous les remercions pour croire en nous.

Merci à tous

Karima

Dédicace

 Je dédie ce mémoire en priorité à mes parents qui m'ont toujours guidé et orienté 
et en particulier pour leurs encouragements  durant tout mon cursus universitaire.

Je dédie également ce mémoire à mes chères frères Abdalh, Mohamed azize, ma
sœur habiba, pour votre aide, vos encouragement si précieux, Votre soutien.

A mes amis, Bouali soumia, hadjer

A mes collègues et professeurs de l'ENSH

A tous ceux qui m'ont soutenue au cours de la réalisation de ce mémoire.



KARIMA

ملخص

التكلفة الرأسمالية لشبكة توزيع المياه بما في ذلك تكلفتها الأولية وتكلفة الإصلاح و الصيانة و التشغيل ضخمة، و لهذا السبب يبحث الباحثون عن طرق جديدة للحصول على تصاميم إقتصادية أكثر، ومع ذلك، فإن تحديد التصميم الأمثل لشبكة توزيع المياه معقدة للغاية.

تتناول هذه الدراسة فقط تحسين أنابيب شبكة توزيع المياه باستخدام برنامج التحسين الذي يربط بين خوارزمية التطور المعقد (SCE) مع برنامج المحاكاة EPANET

الكلمات المفتاحية : خوارزمية التطور المعقد ؛ برنامج التحسين ؛ برنامج المحاكاة

Abstract

The capital cost of a water distribution network including its initial cost, cost of repair, maintenance, and operation is huge; that's why researchers are searching for new methods to obtain more economical designs. However, the determination of the optimal design of a water distribution network is very complex.

this study deals only with the optimization of pipes of water distribution networks using an optimization program that couples the shuffled complex evolution algorithm (SCE) with EPANET simulation software.

Keywords: Shuffled Complex Evolution; Optimization program ; EPANET

Resume :

Le coût en capital d'un réseau de distribution d'eau, y inclus son coût initial, les coûts de réparation, d'entretien et d'opération est énorme ; c'est pourquoi les chercheurs cherchent de nouvelles méthodes pour obtenir des concepts plus économiques. Cependant, la détermination de la conception optimale d'un réseau de distribution d'eau est très complexe.

cette étude ne traite que de l'optimisation des canalisations des réseaux de distribution d'eau à l'aide d'un programme d'optimisation qui couple l'algorithme de évolution complexe shuffled (SCE) avec le logiciel de simulation EpaNet.

Mots-clés: Évolution complexe shuffled; programme d'optimisation; EPANET

SOMMAIRE

INTRODUCTION GENERALE :.....

Chapitre -I- SYNTHESE BIBLIOGRAPHIQUE

I. Introduction :.....	1
I.1 TECHNIQUES DANS LA CONCEPTION DES RÉSEAUX :.....	1
I.1.1 Techniques conventionnelles :.....	1
I.1.2 Techniques d'optimisation traditionnelles :.....	1
I.1.2.1 : Programmation linéaire.....	1
I.1.2.2:Programmation Non-linéaire	2
I.1.2.3: Programmation dynamique	2
I.1.3 Techniques d'optimisation heuristique :.....	3
I.1.3.1 Utilisation des algorithmes génétiques simple :.....	3
I.1.3.2 La combinaison AG et Epanet (GANET) :.....	4
I.1.3.3 La combinaison AG et ANALIS :.....	5
I.1.3.4 Problème du cliff hamming :.....	5
I.1.3.5 Les algorithmes génétiques (SMGA) :.....	5
I.1.3.6 Algorithme fmGA :.....	5
I.1.3.7 Algorithme de recherche GLOBE :.....	6
I.1.3.8 Technique de conception basée sur l'indice de résilience :.....	6
I.1.3.9 La combinaison de AG et fou (G_FUZZNET) :.....	6
I.1.3.10 Algorithme de SFLA :.....	6
I.1.3.11 Algorithme de ACOA :.....	6
I.1.3.12 La fonction de pénalité :.....	7
I.1.3.13 Le raisonnement flou :.....	7
I.1.3.14 La méthode hybride :.....	7
I.1.3.15 AG développé :.....	8
I.1.3.15.1 La combinaison SCE et AG et le couplage avec EPANET :.....	8
Conclusion:	9

Chapitre -II- ALGORITHME EVOLUTIONNAIRE

II. Introduction :.....	10
II.1 Qu'est-ce que l'algorithme génétique ? :.....	10
II.1.1 Mécanisme de l'algorithme génétique :.....	11
I.1.1.1 Concept des chromosomes :	11
I.1.1.2 Génération :	11
I.1.1.3 Sélection:	12
I.1.1.4 Élitisme :	12
I.1.1.5 Crossover :	12
I.1.1.6 Mutation :	13
II.1.2 Mise en œuvre de l'algorithme génétique :.....	14
II.1.3 Domaines d'application :.....	15
II.1.4 Algorithmes génétiques dans le domaine de l'eau :.....	15
II.2 couplage SCE et EPANET :.....	15
II.2.1 Ce qu'est EPANET:.....	16
II.2.2. Capacité de modélisation :.....	16

II.2.2.1 Capacité pour modélisation hydraulique :	18
II.3.2.2 Capacités pour la Modélisation de la Qualité de l'Eau :.....	18
II.2.3 Étapes de l'utilisation d'EpaNet:.....	19
II.2.4 Modèle de simulation hydraulique :	19
II.2.5 Algorithmes d'analyse hydraulique :	19
II.2.6 SHUFFLED COMPLEX EVOLUTION (SCE) :	19
II.2.6.1 Évolution du complexe compétitif (CCE) :	20
II.2.6.2 Sélection des paramètres dans la méthode SCE :.....	21
II.2.6.3 Critères d'arrêt de l'algorithme SCE :.....	22
II.2.6.4 Exploration et exploitation en SEC :.....	22
II.2.6.5 Avantages du SCE par rapport à l'algorithme de recherche traditionnel :...22	
II.2.7 SCHÉMAPROPOSÉ :	23
II.2.7.1 Liaison d'EPANET avec SCE:.....	23
II.2.7.2 langage PYTHON :.....	25
II.2.7.2.1 La cause de création de python :	25
II.2.7.2.2 :Utilisation de Python :.....	25
II.2.7.2.3 : Pourquoi Python	26
II.2.7.2.4 : Syntaxe Python comparée à d'autres langages.....	27
II.2.7.2.5 : Installer un environnement Python	28
II.2.7.2.6 : Qu'est ce que Anaconda ?.....	28
II.2.7.2.7 : Visuelle studio	29
Conclusion :	31

Chapitre -III-

APPLICATIONS DU SCHÉMA PROPOSÉ POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

III. Introduction :	32
III.1 CONCEPTION ET FORMULATION :	32
III.1.1 Coût du réseau :	32
III.1.2 Fonctions de pénalité :	33
III.1.3 Contraintes :	35
III.1.4 MÉCANISME DE FONCTIONNEMENT DU MODÈLE.....	35
III.1.5 CRITÈRES D'ARRÊT :	36
III.1.6. Fonction objective et les contraintes :	36
III.1.8 ÉTUDES DE CAS :	36
III.1.8.1 Présentation du Réseau d'AEP de la zone d'étude :	36
III.1.8.1 Application de notre programme et interprétation :.....	38
Conclusion :	41

CONCLUSION GENERALE :

LISTE DES TABLEAUX

Chapitre -II- ALGORITHME EVOLUTIONNAIRE

Tab.II.1 : Exemple de codage binaire.....	11
Tab.II.2 : Illustration de l'algorithme génétique, ronde 1	14
Tab.II.3 : Illustration de l'algorithme génétique, ronde 2	14

Chapitre -III- APPLICATIONS DU SCHÉMA PROPOSÉ POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

Tab.III.1 : Les Différents Tronçons Avec Leurs Caractéristiques	37
Tab.III.2 : Coûts des différents diamètres (avant optimisation)	38
Tab.III.3 : le dimension des différents diamètres de réseau (après optimisation).....	40
Tab.III.4 : Pressions à divers nœuds	40
Tab.III.5 : Coûts des différents diamètres (après optimisation)	41

LISTE DES FIGURES

Chapitre -II- ESTIMATION DES BESOINS EN EAU POTABLE DE L'AGGLOMERATION

FigureII.1 : Les cinq niveaux d'organisation d'un algorithme génétique.	10
FigureII.2 : Organigramme de l'algorithme génétique simple.	11
FigureII.3 : Exemple de codage de tuyaux en binairev.	12
FigureII.4 : Opérateur de croisement.	13
FigureII.5 : Opérateur de mutation.	13
FigureII.6 : Espace de travail EpaNet.	16
FigureII.7 : Organigramme de l'algorithme SCE.	23
FigureII.8 : Organigramme de la stratégie CCE de l'algorithme SCE.	24
FigureII.9 : Évolution du complexe de Shuffled : Complexes et sous-complexes.	24
FigureII.10 : Schéma de remplacement de solution en méthode Simplex.	25
FigureII.11 : Modèles d'optimisation et de simulation.	25
FigureII.12 : interface de langage python.	27
FigureII.13 : LOGO D'ANACONDA.	28
FigureII.14 : Les versions python installer.	28
FigureII.15 : Installation de ANACONDA.	29
FigureII.16 : Les Déférente IDE dans ANACONDA	29
FigureII.17 : Interface de visuel studio	29

Chapitre -III- CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

Figure III.1 : Organigramme du problème de conception.	34
Figure III.2 : Réseau d'AEP de notre zone d'étude	37
Figure III.3 : interface de programme d'optimisation (GANEO).	38
Figure III.4 : Excution du programme	39

INTRODUCTION GENERALE

INTRODUCTION GENERALE

Le réseau de distribution d'eau est une infrastructure hydraulique composée des divers éléments comme des tuyaux, des réservoirs, des pompes et des vannes. Tous sont essentiels pour fournir de l'eau de qualité acceptable avec une pression spécifiée.

L'eau est fournie au consommateur par un réseau de distribution pour les usages domestiques, commerciaux, industriels et d'irrigation. D'énormes sommes d'argent sont donc allouées à la conception d'un réseau. Il est donc essentiel d'étudier et d'établir un réseau fiable qui réponde aux conditions suivantes (McGhee, 1991) : (1) maintenir la norme de qualité de l'eau dans les conduites de distribution ; (2) établir une conception et une disposition économiques ; (3) fournir une quantité d'eau suffisante ; (4) maintenir la pression hydraulique requise ; (5) assurer une alimentation fiable pendant n'importe quelle période ; (6) assurer à chaque consommateur la qualité de l'eau durant les réparations du système.

La solution d'un problème de réseau est complexe. La nécessité de surmonter ces complexités et de modéliser de façon réaliste les raccords, les vannes, les pompes, les réservoirs de stockage, les variations spatiales et temporelles de la demande en eau et les variations de la qualité de l'eau fait en sorte de développer un modèle de simulation de réseau de distribution d'eau.

Un modèle de simulation de réseau de distribution représente un système réel utilisant des formulations mathématiques pour prédire les réponses du système dans une vaste gamme des conditions sans perturber les systèmes réels.

La simulation du comportement hydraulique d'un réseau de canalisations dans lequel l'eau sous pression est alimentée n'est pas une tâche facile. Il s'agit de résoudre un ensemble d'équations non linéaires simultanées, par exemple l'équation de continuité (conservation du débit à satisfaire à chaque nœud), l'équation d'énergie et l'équation qui relie le débit et les pertes de charge des conduites, telles que les équations de Hazen-Williams, Darcy-Weisbach et Mannings. Avec l'avènement de la technologie de l'informatique douce, les chercheurs se sont intéressés à utiliser cette technologie dans les problèmes avec des calculs itératifs. Il existe de nombreux programmes informatiques utiles et efficaces pour la simulation des réseaux de distribution d'eau. EPANET (Rossman, 1993) est l'un de ces outils de simulation populaires qui joue un rôle important dans la disposition, la conception et le fonctionnement du réseau. Les ingénieurs hydrauliciens utilisent ce modèle de simulation pour déterminer la taille optimale (moindres coûts) des canalisations pour l'alimentation en eau des consommateurs ou pour déterminer les paramètres optimaux du réseau (coefficients de rugosité des canalisations et demandes nodales) pour augmenter la fiabilité du modèle. L'approche traditionnelle utilise une méthode d'essai et d'erreur il prend du temps parce que la détermination des solutions de rechange aux décisions économiques nécessite des ajustements répétitifs des variables (p. ex., diamètres) sur la base des résultats hydrauliques obtenus à partir d'un modèle de simulation (Epanet), jusqu'à ce que certaines spécifications prédéfinies (pression nodale minimale requise) soient respectées. De plus, la complexité de la procédure d'essai et d'erreur augmente de façon exponentielle avec le nombre des variables de décision, les modifications proposées et les conditions de fonctionnement correspondantes (Wu et al., 2001). Pour remédier à ces difficultés, il est possible de recourir à une technique d'optimisation qui recherche de manière exhaustive une solution optimale dans le domaine de l'espace disponible et réalisable.

Une technique d'optimisation utilise une formulation mathématique qui décrit les réponses du système pour différents paramètres de conception. La fonction objective et les contraintes sont introduites pour contrôler l'ensemble de la stratégie. Cette technique génère automatiquement les variables en utilisant plusieurs approches déterministes et probabilistes et le rendement est déterminé par la fonction objective. ont décrit que la conception optimale du réseau est assez compliquée en raison de la relation non linéaire entre le débit et la perte de charge et de la présence de variables discrètes, comme la taille du marché des conduites. De

plus, la fonction objective, qui représente le coût du réseau, est non linéaire, non convergente et non convexe. Gessler (1985) a également décrit ce qui accroît les difficultés associées à la conception du réseau de canalisations :

1. Les dimensions des tuyaux ne sont pas variables de façon continue parce qu'elles doivent être choisies à partir d'une liste des tailles disponibles. De nombreuses techniques d'optimisation tiennent compte des variables continues; cependant, après la fin du programme, l'arrondissement des diamètres continus peut conduire à une solution non optimale ou même une solution impossible;
2. La fonction objective qui représente le coût du réseau est arbitraire. L'approximation mathématique peut conduire à des résultats incorrects ;
3. La fonction objective d'un réseau maille peut comporter plusieurs minima locaux ; et Les besoins en pression peuvent varier selon l'évolution du scénario de la demande (débit de pointe, débit moyenne et débit en cas d'incendie). Plusieurs scénarios de la consommation doivent être pris en compte dans l'optimisation.

OBJECTIFS DE L'ÉTUDE

Cette étude utilise un algorithme évolutif, Shuffled Complex Evolution (SCE) dans le domaine de la modélisation des réseaux de distribution d'eau. On construit un modèle d'optimisation qui couple une technique d'optimisation, Shuffled Evolution Complex (SCE) et un modèle de simulation, EPANET, basée sur la technique de Nelder et Mead de la recherche par l'algorithme du simplex. La méthode est utilisée pour avoir un réseau de distribution fiable.

L'objectif de cette étude est d'explorer et d'améliorer la modélisation des réseaux de distribution d'eau. Les principaux objectifs sont les suivants :

1. Proposer et appliquer un algorithme fiable, Shuffled Complex Evolution (SCE), pour optimiser la conception des réseaux de distribution d'eau. SCE est couplé à un solveur de réseau hydraulique EPANET ;
2. Réhabilitation d'un réseau de distribution d'eau existant.
3. Faire la comparaison avant et après optimisation.

ORGANISATION DE LA THÈSE

Le chapitre I décrit les travaux de recherche antérieurs sur les méthodes d'optimisation.

Le chapitre II Nous avons parlé sur Les algorithmes génétiques et leurs principes d'opération, ainsi que leurs applications dans le domaine d'AEP. Et aussi Le solveur de réseau EPANET et Shuffled Complex Evolution (SCE) sont décrits en détail dans la section, La procédure de liaison entre la SCE et EPANET est également présentée dans ce chapitre.

l'application du modèle proposé dans la conception de réseaux de distribution d'eau de la commune de SIDI LAKHDAR est décrits au chapitre III Enfin, les conclusions de la présente étude.

Chapitre -I-

SYNTHESE BIBLIOGRAPHIQUE

I Introduction

Un réseau de distribution d'eau bien planifié est essentiel au développement des zones urbaines. Le réseau est conçu pour répondre aux diverses demandes des consommateurs tout en satisfaisant les exigences minimales de pression à certains nœuds. Au stade de la conception, il est intéressant de trouver les solutions les moins coûteuses qui répondent à un ensemble des contraintes, y compris la demande et les exigences de pression. Souvent, il est également intéressant d'arriver à des solutions moins coûteuses qui, cependant, violent légèrement les contraintes. De nombreux problèmes d'optimisation dans divers domaines ont été résolus à l'aide de divers algorithmes d'optimisation. Les techniques d'optimisation traditionnelles telles que la programmation linéaire (LP), la programmation non linéaire (NLP) et la programmation dynamique (DP) ont joué un rôle majeur dans la résolution de ces problèmes. Cependant, leurs inconvénients génèrent une demande pour d'autres types d'algorithmes[1], Par conséquent, les intérêts de la recherche se sont concentrés sur le développement d'algorithmes évolutifs efficaces (est détaillé dans chapitre II) pour rechercher la combinaison optimale des variables de décision (p. ex., diamètres des tuyaux). à partir d'un grand nombre des solutions.[2]

Dans ce chapitre, différentes techniques connues dans la conception des réseaux de distribution d'eau sont d'abord passées en revue.

I.1 TECHNIQUES DANS LA CONCEPTION DES RÉSEAUX DE DISTRIBUTION D'EAU

I.1.1 Techniques conventionnelles :

Dans la conception et l'analyse des réseaux de distribution d'eau, la procédure conventionnelle utilise une approche d'essai et d'erreur. Le rendement dépend de l'intuition, de l'expérience, des compétences et des connaissances des utilisateurs. Toutefois, cette approche est inefficace, en particulier dans la conception et l'analyse des grands systèmes complexes. Par exemple, pour déterminer le réseau de distribution d'eau les moins coûteux, il faudrait choisir la taille des tuyaux disponibles sur le marché. Ainsi, un grand nombre de simulations répétitives est nécessaire pour arriver à un réseau satisfaisant.[2]

I.1.2 Techniques d'optimisation traditionnelles :

Plusieurs techniques d'optimisation, au lieu de l'approche par essais et erreurs, ont été utilisées dans la conception des nouveaux réseaux de distribution d'eau ainsi que dans l'extension des réseaux existants. Ces techniques d'optimisation traditionnelles comprennent la programmation linéaire, non linéaire et dynamique. [3]

I.1.2.1 Programmation linéaire :

Alperovits et Shamir (1977) ont d'abord présenté une méthode de gradient de programmation linéaire (GPL) dans la conception optimale du réseau de distribution d'eau. Pour appliquer le GPL, ils ont linéarisé la formulation mathématique. Les longueurs segmentées de la conduite de différents diamètres ont été utilisées comme variables de prise de décision. La fonction objective était de minimiser le coût de la longueur totale des tuyaux. Toutefois, il n'est pas souhaitable d'avoir des tuyaux qui changent constamment de taille le long du réseau. Une telle disposition des tuyaux cause un goulot d'étranglement dans le système lorsque la direction de l'écoulement change (Walski et al., 1990). Plusieurs chercheurs ont par la suite amélioré la méthode du GPL (Quindry et al., 1981; Morgan et Glulter, 1985; Fujiwara et al., 1987; Kessler et Shamir, 1989; Fujiwara et Khang, 1990; et Eiger et al., 1994).[3]

Les approches améliorées utilisaient des processus **itératifs**. Les débits ou les hauteurs manométriques sont fixes et les diamètres de tuyauterie sont optimisés en fonction des exigences de débit et de pression spécifiées. Morgan et Glulter (1985) ont tenu compte des multiples modèles de demande dans leurs travaux. Ils ont adapté un modèle de programmation linéaire avec un solveur de réseau Hardy-Cross pour concevoir des systèmes d'alimentation en eau. Kessler et Shamir (1989) ont présenté une méthode de gradient de programmation linéaire (GPL) en deux étapes. Dans un premier temps, pour un ensemble donné des débits, plusieurs ensembles des hauteurs de chute (perte de charge) sont déterminés

par LP. Dans la deuxième étape, les flux sont modifiés en fonction du gradient des fonctions objectives. Eiger et al. (1994) ont par la suite utilisé la même formulation que Kessler et Shamir (1989). Ils ont résolu le problème en utilisant une branche non lisse et des algorithmes liés, et la théorie de la dualité. Les algorithmes sont une combinaison de processus primaires, doubles et arrêtés lorsque l'écart entre la meilleure solution et la limite inférieure globale se situe dans une tolérance prescrite. Le modèle LP subit des pertes dans le processus en raison de la linéarisation de problèmes réels non linéaires. Il n'est pas toujours pratique de linéariser le problème et dans certains cas, cela peut perturber la solution du problème initial.[1]

I.1.2.2 Programmation Non-linéaire :

Une technique de programmation non linéaire (PNL) a été mise au point par Chiplunkar et al (1986). Su et al (1987) ; Lansey et Mays (1989) ; et Duan et al (1990) ont appliqué la PNL pour optimiser la conception des réseaux de distribution d'eau. Comparativement au modèle LP, le modèle NLP peut traiter de multiples modèles de demande et un nombre beaucoup plus élevé des variables de conception. Chiplunkar et al. (1986) ont utilisé la méthode de Davidon Fletcher-Powell, et ont incorporé la méthode de Newton à la méthode de la théorie graphique.[4]

Cependant, dans la technique PNL, les flux de Bouclesa sont assignés comme l'ensemble de variables optimal qui est recherché pour minimiser le coût des systèmes. Un des problèmes signalés est qu'un réseau en boucle devient souvent un réseau d'arbres avec plusieurs valeurs nulles dans certains diamètres de tuyaux. De plus, ils ont constaté que le modèle PNL convergeait souvent prématurément vers les **minimums** locaux. Su et al. (1987) ont utilisé le PNL fondé sur la technique du gradient réduit généralisé (GRG) et un modèle de simulation à l'état d'équilibre pour concevoir les variables de contrôle optimales. Lansey et Mays (1989) ont déterminé les variables de décision en utilisant le même optimiseur PNL et un modèle de simulation de réseau hydraulique, KYPIPE (Wood, 1980). [4]

Au cours des dernières décennies, les algorithmes de programmation non linéaires sont devenus l'une des méthodes les plus utilisées pour résoudre les problèmes des réseaux de distribution d'eau. Les plus efficaces de ces méthodes sont les algorithmes basés sur les gradients qui nécessitent au moins les dérivées de premier ordre des fonctions objectives et des fonctions de contrainte ; celles-ci sont nécessaires pour définir la direction de recherche appropriée. Les techniques basées sur les gradients permettent d'identifier facilement l'optimum relatif le plus proche de la conception optimale. Cependant, ces méthodes ne garantissent pas la solution globale optimale si l'espace de conception est non convexe.[4] Elle est également inadéquate dans les cas où l'espace de conception est discontinu, car la dérivée de la fonction objective et des contraintes peut devenir singulière au-delà de la limite de la discontinuité. De plus, les diamètres des tuyaux pris en considération dans le PNL sont continus et peuvent ne pas correspondre aux diamètres des tuyaux commerciaux disponibles (Les diamètres de tuyaux disponibles sur le marché ne sont certainement pas continus.) et nécessitent l'arrondissement de la solution finale. Cet arrondissement de la solution détruit la qualité de la solution optimale, même s'il ne garantit pas une solution réalisable (Savic et Walters, 1997; Gupta et al., 1999; Cunha et Sousa, 1999; Simpson et al., 1994). Le PNL ne peut pas non plus gérer les grands réseaux de distribution d'eau. Récemment, le PNL n'est pas largement accepté dans le domaine de l'optimisation. [3]

I.1.2.3 Programmation dynamique :

Depuis les années soixantes, la programmation dynamique (DP) (Wong et Larson, 1968), une technique mathématique, a été adaptée à divers problèmes d'optimisation liés à l'ingénierie et à la gestion des ressources en eau. Il résout des problèmes d'optimisation complexes en les divisant en une série de sous-problèmes. Ce sous-problème est appelé étape. La PD s'applique aux problèmes où chaque étape est liée à l'étape précédente. L'état de sortie d'un étage est considéré comme l'état d'entrée de l'étage suivant.[5] La programmation dynamique peut être utilisée pour l'optimisation du réseau distribution ramifié où un étage est

le pipeline (Oléoducs c'est la longueur de réseau) de conception entre les nœuds et l'état représente la pression d'un nœud. Les décisions prises à chaque étape produisent un coût de retour qui sert à calculer le coût du réseau. Il peut prendre en charge une forme plus générale de fonction de coût qui à son tour, dépend de la pression et du diamètre de la conduite.[5]

Vamvakeridou-Lyroudia (1993) a présenté une approche de programmation dynamique en deux étapes avec un algorithme heuristique. Dans un premier temps, la taille des tuyaux a été initialisée par une technique heuristique. Dans un deuxième temps, DP a été appliqué pour calculer de manière itérative la solution optimale. A chaque étape, le modèle de simulation de réseau Newton-Raphson a été utilisé. Ils ont développé LOOPOPOPT sur la base des techniques mentionnées ci-dessus pour résoudre les problèmes de réseau de distribution d'eau de type en boucle, ramifié ou mixte.[5]

Lall et Percell (1990) ont mis au point un optimiseur de programmation dynamique (GPO) dans les réseaux de gazoducs. En tant qu'outil d'optimisation, il a été utilisé pour déterminer la stratégie d'état stable réalisable dans l'exploitation des stations de compression satisfaisant à plusieurs contraintes afin de réduire au minimum le coût global du carburant pour l'exploitation du pipeline. Toutefois, l'application de la programmation dynamique était limitée à système réseau. Ils ont observé que si la taille du système augmente, le temps de calcul nécessaire pour résoudre la stratégie optimale devient très important.[4]

I.1.3 Techniques d'optimisation heuristique :

Gessler (1985) et Loubser et Gessler (1993) ont appliqué l'approche du dénombrement à la conception et à la réhabilitation du réseau de distribution d'eau. Dans la technique de numération, le modélisateur attribue un espace de recherche en spécifiant une gamme des diamètres commerciaux pour chaque conduite du réseau. L'algorithme tient compte de toutes les combinaisons possibles de diamètres de tuyaux et vérifie si les contraintes de pression sont respectées. Finalement, les algorithmes choisissent la combinaison des tuyaux la moins coûteuse. Cette combinaison optimale répond bien entendu aux contraintes spécifiées. Cet algorithme élimine certaines complexités. Cependant, l'inconvénient le plus important est qu'il faut beaucoup de temps de calcul pour trouver même une solution sous-optimale. Ceci est dû au fait que le tube individuel est dimensionné en fonction des dimensions discrètes disponibles sur le marché et que l'ensemble des combinaisons possibles d'options de décision contient certaines solutions de qualité inférieure. La vérification de toutes ces solutions nécessitera un temps de calcul élevé.[1]

La nature discrète des variables de décision (diamètre des tuyaux) limite la recherche à un nombre fini des diamètres possibles. Toutefois, dans la pratique, ce nombre est très élevé pour les grands réseaux. L'évaluation de chaque conception possible n'est pas pratique pour la plupart des problèmes. L'évaluation de certaines conceptions à l'aide d'algorithmes évolutifs rend cette approche acceptable. Récemment, les chercheurs se sont concentrés sur l'utilisation de techniques méta- heuristiques pour évaluer certaines conceptions de réseaux.[2]

Ces techniques de recherche résolvent normalement le problème d'optimisation sans contrainte et sélectionnent les diamètres de tuyauterie (variables de conception) à l'intérieur des gammes prédéfinies. Ainsi, les contraintes de conception sont automatiquement satisfaisantes. La technique d'optimisation heuristique fonctionne avec un solveur de réseau qui traite les relations hydrauliques et les contraintes. Si les exigences hydrauliques ne sont pas satisfaites, la fonction objective est modifiée en ajoutant une valeur numérique élevée ou un terme de pénalité (correction) pour arrêter la recherche près de ce point ou dans cette direction (Savic et Walters, 1997).[2]

I.1.3.1 Utilisation des algorithmes génétiques simple :

Les approches méta-heuristiques actuellement utilisées comprennent les algorithmes génétiques (Goldberg et Kuo, 1987 ; Simpson et al., 1994 ; Savic et Walters 1997 ; Dandy et al. 1996 ; Gupta et al, 1999 ; Wu et Simpson, 2001), le recuit simulé (Cunha et Sousa, 1999), les algorithmes des feuilles des grenouilles mélangées (Eusuff et Lansey, 2003), les algorithmes d'optimisation des colonies de fourmis (Maier et al., 2003) qui traitent simultanément la population des points dans la recherche d'un optimum global, La stratégie

de recherche qui est basée sur la fonction objective, améliore la qualité de la solution à travers des processus évolutifs.[1]

Simpson et ses collaborateurs. (1994) ont utilisé des algorithmes génétiques simples dans lesquels chaque population individuelle est représentée dans une chaîne des bits de longueur identique qui encode une solution possible. Toutes les populations codées binaires des points (chromosomes) subissent trois opérations : sélection, croisement et mutation. Le GA simple utilise la sélection de la roulette, le crossover d'un point et la mutation du bit wise pour déterminer la conception optimale du réseau. Un solveur de réseau à l'état stable est utilisé pour calculer la performance hydraulique de chacun des réseaux de la population. Si le réseau n'est pas suffisant pour répondre aux contraintes de la hauteur d'élévation, les insuffisances sont intégrées dans la fonction objective (coût du réseau) comme pénalités pour le calcul du coût total du réseau. Par conséquent, la qualité de chaque solution est évaluée à l'aide d'une fonction de fitness qui est l'inverse du coût total du réseau. La performance de l'AG est comparée à l'énumération complète et aux techniques d'optimisation de programmation non linéaire sur un problème typique. Simpson et al. (1994) ont constaté que l'AG avait obtenu une solution globale dans un nombre relativement peu d'évaluations des fonctions.[6]

L'AG simple a été améliorée par Dandy et ses collaborateurs (1996) à l'aide du concept d'échelle de puissance variable de la fonction fitness, d'un opérateur de mutation adjacente et des codes gris. La puissance du fitness (inverse de la valeur de la fonction d'objectif) peut augmenter par étapes au fur et à mesure que la course GA se développe. Une faible valeur d'exposant est employée (utilisée) au début de l'AG, ce qui préserve une certaine diversité de population et l'exploration globale de l'espace de solution dans la première génération.[2]

Une valeur élevée d'exposant est nécessaire pour renforcer les petites différences dans la chaîne de la fonction fitness. En outre, ils ont introduit l'opérateur de mutation rampante (glissante) ou d'adjacence avec l'opérateur de mutation binaire couramment utilisé. L'opérateur de mutation d'adjacence est appliqué à une variable de décision complète sélectionnée au hasard dans la chaîne codée. Cet opérateur fait muter (transforme) la sous-chaîne des variables de décisions sélectionnées en une sous-chaîne des variables de décisions adjacentes vers le haut ou vers le bas dans la liste des choix de variables de conceptions. Enfin, au lieu des codes binaires, on utilise des codes gris pour représenter les variables de conception. En raison de ces nouvelles caractéristiques, Dandy et ses collaborateurs (1996) ont conclu que l'AG améliorée avait un meilleur rendement que l'AG simple. Ils ont également fait valoir que l'AG améliorée a donné des meilleurs résultats que les méthodes de programmation linéaire, non linéaire et dynamique et que la méthode de recherche énumérative. [6]

I.1.3.2 La combinaison AG et Epanet (GANET):

Savic et Walters (1997) ont également utilisé la norme GA en conjonction avec le solveur de réseau EPANET. Ils ont développé un modèle informatique GANET pour la conception d'un réseau de distribution à moindre coût. Dans GANET, l'AG standard qui est basé sur l'alphabet binaire d'utilisation génétique naturelle pour générer les chromosomes comme ceux trouvés dans l'ADN. Les chromosomes qui représentent une solution particulière du problème améliorent leur qualité grâce à plusieurs opérateurs génétiques. GANET est facile à utiliser lorsque les données d'entrée sont les mêmes que celles des modèles de simulation hydraulique, en plus des paramètres algorithmiques génétiques. La capacité de GANET à optimiser la conception a été démontrée sur trois problèmes. Pour améliorer la capacité de GANET, des systèmes d'information géographique (SIG) (StruMap), ensemble avec un solveur de réseau (HARP), une base des données relationnelles et une bibliothèque d'algorithmes génétiques orientée objet ont été intégrés (Atkinson et al., 1998). L'ensemble qui en résulte est conçu pour être utilisé avec un large éventail des problèmes d'optimisation

liés à la gestion des réseaux d'eau, comme les problèmes de conception, de réhabilitation et d'étalonnage des réseaux.[7]

I.1.3.3 La combinaison AG et ANALIS

Gupta et al. (1999) ont appliqué AG avec un simulateur hydraulique ANALIS (Bassin et al., 1992) qui était basé sur la théorie des graphes pour évaluer la performance hydraulique de la conception du réseau. Le résultat obtenu a été comparé à la technique PNL. Ils ont constaté que bien que le PNL ait convergé très rapidement, l'AG a fourni de meilleures solutions. Bien que la qualité de la solution finale ait été améliorée, ils ont eu besoin d'un effort de calcul considérable pour arriver au coût de réseau le moins élevé.[8]

I.1.3.4 Problème du cliff hamming :

Toutes les méthodes décrites ci-dessus utilisent un simple code binaire ou gris codé GA pour représenter les diamètres des conduites (variables de conception). Ces schémas de codage produisent parfois des états redondants (la valeur décodée peut ne pas appartenir au domaine des paramètres) qui ne représentent aucun des diamètres des conduites et cliff hamming (Vairavamoorthy et Ali, 2000). Le problème du cliff hamming se produit lorsque le code binaire de deux valeurs adjacentes diffère dans chacun de leurs bits ; la convergence vers l'optimum global peut ne pas être atteinte dans cette condition. Vairavamoorthy et Ali (2000) ont résolu ces problèmes en incorporant un codage réel aux variables et un modèle de fonction de transfert linéaire pour éviter d'avoir à utiliser un solveur de réseau hydraulique. Leur modèle proposé a été testé sur plusieurs problèmes de référence et a prouvé son efficacité par rapport aux méthodes simples basées sur l'AG.[7]

I.1.3.5 Les algorithmes génétiques désordonnés structurés multi-objectifs (SMGA) :

Halhal et al (1997) et Walters et al (1999) proposent une version améliorée de l'AG, appelée algorithmes génétiques désordonnés structurés multi-objectifs (SMGA). Dans SMGA, un codage flexible avec une longueur de chaîne variable est utilisé pour représenter la population des points. Cet algorithme suit un processus évolutif progressif dans lequel les premières longueurs de chaîne sélectionnées des solutions sont autorisées à augmenter à travers le processus de concaténation dans la génération suivante avec l'amélioration de la qualité des points. Le SMGA présente deux avantages majeurs par rapport à l'algorithme génétique standard (AGS) :[13]

- (1) il encode seulement un petit nombre des variables décisionnelles pertinentes ayant une petite longueur de chaîne, tandis que SGA encode toutes les variables décisionnelles même si certaines d'entre elles n'ont pas besoin d'être mises à niveau;
- (2) puisque SMGA ne prend pas en compte toutes les variables décisionnelles, le temps de calcul pour atteindre la solution optimale ou presque optimale est inférieur à SGA.

Prasad et ses collaborateurs (2003) ont décrit une méthodologie d'algorithme génétique multi objective pour la conception optimale d'un réseau de distribution d'eau afin d'obtenir le front pareto-optimal. Prasad et Park (2004) ont présenté une approche algorithmique génétique multi objective pour la conception d'un réseau de distribution d'eau. Les objectifs considérés sont la minimisation du coût du réseau et la maximisation d'une mesure de fiabilité.[5]

I.1.3.6 Algorithme fmGA (first messy genetic algorithm) :

Wu et Simpson (2001) et Wu et al (2001) ont également signalé un premier algorithme génétique désordonné (fmGA) à objectif unique - pour améliorer l'efficacité d'une procédure d'optimisation pour la conception optimale des réseaux de distribution d'eau. Le premier GA en désordre résout le problème du réseau de distribution d'eau en utilisant deux boucles, à savoir, la boucle intérieure et la boucle extérieure. La boucle externe initialise une grande taille de population des variables aléatoires. La boucle intérieure se compose d'une phase de filtrage en blocs de construction et d'une phase de juxtaposition. La phase de filtrage par blocs de construction filtre toutes les chaînes de caractères de la population pour sélectionner les chaînes courtes les mieux adaptées et réduit également la longueur des

chaînes. Et la phase de juxtaposition produit une nouvelle chaîne de la population en utilisant des opérateurs génétiques communs (sélection et reproduction) pour aller vers une solution optimale. Enfin, la fmGA a produit des résultats raisonnables pour les problèmes de référence typiques.[6]

I.1.3.7 Algorithme de recherche GLOBE :

Au lieu d'utiliser un algorithme d'optimisation unique, Abebe et Solomatine (1998) ont appliqué GLOBE (Solomatine, 1995) qui comprend plusieurs algorithmes de recherche dont AG. Ils ont constaté que très peu d'algorithmes parviennent à des solutions optimales ou presque optimales. Cunha et Sousa (1999) ont introduit un algorithme de recherche aléatoire (recuit simulé) qui est basé sur l'analogie avec le processus de recuit physique avec la méthode de recherche Newton pour résoudre l'équation de réseau.[9]

I.1.3.8 Technique de conception basée sur l'indice de résilience :

Todini (2000) a présenté une nouvelle technique de conception des réseaux de distribution d'eau basée sur l'indice de résilience pour améliorer la fiabilité et la disponibilité de l'eau en cas de défaillance. Dans cette technique, le concept de résilience est utilisé pour développer une approche heuristique d'optimisation qui traite des coûts par rapport à l'espace de résilience, du bord des solutions non dominées ou de l'ensemble Pareto, La résilience du réseau a été utilisée comme mesure de fiabilité pour fournir une pression supérieure à la pression minimale autorisée aux nœuds. La technique proposée a été testée sur un grand réseau de distribution d'eau.[9]

I.1.3.9 La combinaison de AG et fou (G_FUZZNET):

Vamvakeridou-Lyroudia (2001) a développé G_FUZZNET qui est une combinaison de flou et de GA pour la conception de ce réseau. Ils ont évalué les fonctions d'appartenance d'adhésion agrégées totales pour chaque solution possible. Il y avait trois étapes d'agrégation:[10]

- 1) l'agrégation pour chaque nœud : les contraintes de pression pour les nœuds sont agrégées;
- 2) l'agrégation pour chaque liaison : les fonctions d'appartenance partielle pour les contraintes de vitesse pour les liaisons sont agrégées ;
- 3) un agrégateur de pression pour tous les nœuds : l'agrégateur de pression totale est appliqué pour évaluer la fonction de membre totale du système.

I.1.3.10 Algorithme de SFLA (shuffled frog leaping algorithm) :

Eusuff et Lansey (2003) ont proposé un algorithme de saut de grenouille en marche (SFLA), c'est un algorithme méta-heuristique basé sur l'évolution memétique (transformation des grenouilles) et l'échange d'informations entre la population. Les grenouilles qui sont les hôtes des mèmes (se composent de mémotype comme le gène dans le chromosome dans les algorithmes génétiques) recherchent la particule contenant le plus de nourriture dans un marais en améliorant leurs mèmes. L'amélioration des mèmes se produit en partageant l'information entre les grenouilles. Alternativement, les mèmes qui représentent les coordonnées des grenouilles peuvent changer leur position par l'ajustement du mémotype. Le saut des grenouilles améliore leur performance memétique en utilisant le processus impliqué dans l'optimisation de l'essaim des particules (PSO) dans un sous-ensemble des grenouilles virtuelles (memplex). Les memplexes sont finalement mélangés pour aller vers l'optimum global. Ils ont obtenu la solution optimale et presque optimale avec le coût d'un temps de calcul plus élevé.[10]

I.1.3.11 Algorithme de ACOA (ant colony optimization algorithms)

Maier et ses collaborateurs (2003) ont utilisé un algorithme évolutif, c'est-à-dire des algorithmes d'optimisation des colonies de fourmis (ACOA) qui fonctionnent sur la base du comportement de recherche de nourriture des fourmis. Dans la conception des réseaux de distribution d'eau, les points de décision sont choisis parmi les diamètres de conduites

disponibles. Ces choix dépendent de certaines de la fourmi et des valeurs heuristiques. La valeur heuristique est l'inverse du coût total de chaque choix. Les caractéristiques des fourmis sont modifiées de manière à favoriser le choix, ce qui réduit le coût du réseau. De même, les caractéristiques sont diminuées si le choix du réseau ne répond pas aux contraintes de pression.[6]

I.1.3.12 La fonction de pénalité :

Dans la méthode de solution par technique d'optimisation heuristique, la faisabilité du membre individuel dépend du degré de violation de contrainte ou la distance de la région réalisable qui est comptabilisée en utilisant une fonction de pénalité. La fonction de pénalité tient compte principalement du déficit de pression nodale dans la conception du réseau de distribution d'eau. Elle est proportionnelle au déficit de pression maximale (différence maximale entre la hauteur piézométrique requise et la hauteur piézométrique simulée de chaque point) multiplié par un facteur de pénalité. La plupart des méthodes ne tiennent compte que de la violation de pression maximale et ignorent l'importance du degré de violation à d'autres nœuds. Si les déficits de pression maximale sont les mêmes, des pénalités égales seront imposées pour les conceptions où la majorité des nœuds violent leurs contraintes de pression et celles où seulement quelques nœuds sont en violation de pression (Vairavamoorthy et Ali, 2000). De plus, le facteur de pénalité dans la fonction est très important et doit être réglé pour obtenir efficacement des résultats optimaux. Si le facteur de pénalité est trop faible, des nombreuses solutions irréalisables domineront la population algorithmique. Si le facteur de pénalité est très élevé, les bonnes solutions qui viennent d'échouer seront éliminées définitivement dans le processus de recherche (Wu et Simpson, 2002).[11]

Les méthodes habituelles utilisent l'approche par essais et erreurs pour sélectionner les coefficients de pénalité appropriés et le facteur de pénalité fixe est utilisé dans chaque cycle, quel que soit le degré de violation de la pression et le nombre des nœuds de demande pour lesquels la pression est inférieure à la pression minimale requise. Vairavamoorthy et Ali (2000) ont proposé de considérer un facteur de pénalité variable en fonction du degré de violation. Ils ont additionné le déficit de pression à tous les nœuds où la pression nodale ne répondait pas à l'exigence minimale (le n'a pas atteint le minimum requis). Eusuff et Lansey (2003) ont également suivi la même approche pour calculer le coût de la pénalité.[11]

I.1.3.13 Le raisonnement flou :

Vamvakeridou-Lyroudia (2001) a suggéré un raisonnement flou pour le traitement des contraintes et les vérifications de faisabilité. Au lieu des fonctions de pénalité, il a utilisé la fonction d'adhésion totale agrégée à l'ensemble des solutions faisable. Une fonction d'appartenance non linéaire en forme de S a été employée. Trois contraintes sous forme des contraintes de diamètre, de vitesse et de pression ont été introduites. Les fonctions d'adhésion pour chacune de ces contraintes ont été incorporées pour les ajouter au coût réel du réseau.

Wu et Simpson (2002) ont introduit une approche appelée stratégie de recherche de limites auto-adaptative dans un algorithme génétique rapide et complexe pour surmonter l'approche de sélection d'essai des coefficients de pénalité. La méthode adapte et co-évolue le facteur de pénalité pour rechercher la limite de la région réalisable et irréalisable.[11]

I.1.3.14 La méthode hybride :

Zyl et ses collaborateurs (2004) ont mené une étude pour améliorer l'efficacité de l'optimisation opérationnelle de l'AG par une méthode hybride qui combine la méthode de l'AG avec une stratégie de recherche de grimpeur. Les stratégies de grimpeur complètent les AG en étant efficaces dans la recherche d'un optimum local. Deux stratégies de grimpeur, les méthodes Hooke et Jeeves et Fibonacci, ont été étudiées. La méthode hybride s'est avérée supérieure à l'AG classique pour trouver rapidement une bonne solution, à la fois lorsqu'il est appliqué à un problème d'essai et à un grand réseau de distribution d'eau existant.[11]

I.1.3.15 AG développé :

Konak et ses collaborateurs (2006) ont présenté un aperçu et un tutoriel décrivant les algorithmes génétiques développés spécifiquement pour les problèmes. Avec des objectifs multiples. Artem V Babyan et ses collaborateurs (2007) se sont concentrés sur la résolution du problème de la conception robuste des réseaux de distribution d'eau à moindre coût, dans l'incertitude des paramètres d'entrée en gardant l'objectif de minimiser le coût total de conception sous réserve d'un niveau cible de robustesse du système.[8]

Van et al (2008) ont présenté une nouvelle approche pour déterminer une valeur de pénalité en fonction du degré de défaillance, des critères de pression définis et de l'importance de la liaison fournissant un nœud spécifique. D'autres modifications sont également apportées aux procédures de croisement et de mutation pour assurer une augmentation de la convergence des algorithmes. Prasad (2008) a établi un paramètre pour évaluer les variations de la satisfaction de la demande aux différents nœuds d'un réseau de distribution d'eau potable et a développé un modèle de conception optimale et fiable d'un système d'approvisionnement en eau adapté aux conditions indiennes en utilisant des algorithmes génétiques.[12]

Afshar (2009) a proposé un algorithme génétique compact pour réduire les besoins de stockage et de calcul des algorithmes génétiques basés sur les populations. Chandramouli (2015) a utilisé GA comme outil d'optimisation et EPANET comme solveur hydraulique pour la conception fiable des réseaux de distribution d'eau dans l'environnement MATLAB.

Il ressort clairement de ce qui précède que des nombreux chercheurs ont développé de différentes approches pour une conception optimale des réseaux de conduites d'alimentation en eau, en particulier avec GA. Mais aucun d'entre eux ne prévoit une procédure de conception détaillée.[12]

Vasques et al[14]ont appliqué l'AG à la méthode de fiabilité du premier ordre (FORM) pour estimer la probabilité de panne du système sous une allocation donnée de charge de déchets afin de trouver des solutions de répartition entre coût et fiabilité du traitement.

I.1.3.15.1 La combinaison SCE et AG et le couplage avec EPANET :

Récemment, les techniques d'optimisation sont devenues largement utilisées dans les applications de modélisation informatique. Un avantage majeur de l'application des techniques d'optimisation avec les modèles est d'obtenir de meilleurs résultats sans effort ni temps significatif.

Les techniques d'optimisation peuvent être divisées en deux groupes : les méthodes de recherches globales telles que l'algorithme génétique (GA)et l'algorithme Shuffled Complex Evolution (SCE), et les méthodes de recherche locale, telles que la méthode simplex, la méthode de recherche pattern, et la méthode directionnelle. Des nombreuses études ont rapporté que les méthodes de recherche globale ont donné des meilleurs résultats que les méthodes de recherche locale.

Des nombreuses études ont révélé que l'algorithme SCE-UA fournissait des meilleures estimations de la solution comparative à d'autres procédures de recherche mondiales et locales pour le calibrage des modèles. Cooper et al[15]ont évalué trois méthodes d'optimisation globale, dont la SCE-UA, GA et le recuit simulé (SA). Ils ont constaté que la performance de l'optimisation était influencée par la fonction objective et la position de départ de la recherche d'optimisation, et que la méthode SCE donnait des meilleurs résultats que les autres méthodes. Franchini et al[16]ont rapporté que les solutions de SCE-UA étaient stables mais caractérisées par de nombreuses valeurs de paramètres fixées à la limite de leur plage potentielle. Kuczera[17]a indiqué la raison pour laquelle le SCE est meilleur que GA en raison du partage périodique de l'information entre les recherches locales en simplex. Thyer et al[18]ont révélé que l'utilisation par SCE-UA de multiples complexes et le mélange permettait une recherche plus efficace de l'espace des paramètres que le simplex unique de GA.

Ngo et al[19]ont proposé d'optimiser les stratégies de contrôle pour l'exploitation des réservoirs en appliquant les méthodes SCE. Kerachian et Karamouz[20]ont identifié des règles d'exploitation pour la gestion de la qualité de l'eau dans les systèmes réservoir-fleuve à l'aide d'une technique stochastique de résolution des conflits basée sur l'AG.

Des nouveaux algorithmes d'optimisation ont été développés et certains algorithmes ont démontré une meilleure capacité que GA ou SCE pour certains problèmes [21]. Toutefois, les algorithmes GA et SCE sont utilisés pour le calibrage automatique.

Le SCE a été développé au sein du Département d'hydrologie et des ressources en eau de l'Université de l'Arizona et combine la procédure simplex avec le concept de recherche aléatoire contrôlé, l'évolution compétitive [23] et un concept de complexe shuffling [22].

Goldberg a indiqué qu'une AG diffère des méthodes de recherche traditionnelles de la manière suivante:[12]

- ✓ Les AG fonctionnent avec le codage du jeu des paramètres, pas avec les paramètres eux-mêmes.
- ✓ Recherche de la population des points, pas un seul point.
- ✓ Les AG utilisent des informations sur les fonctions objectives (AGins) et non sur des dérivés ou d'autres connaissances auxiliaires pour déterminer l'adéquation de la solution.
- ✓ Les AG utilisent des règles de transition probabilistes et non des règles déterministes.

Conclusion :

Afin de faciliter la tâche des jeunes chercheurs et étudiants, une procédure de conception détaillée utilisant le solveur EPANET avec des algorithmes génétiques dans l'environnement Jupyter et visuel studio) avec langage PYTHON pour une conception optimale basée sur la fiabilité des réseaux d'alimentation en eau ; une tentative a été faite dans la présente étude. Les fonctions de la boîte à outils EPANET sont les mieux adaptées à l'analyse d'un réseau des canalisations et on peut facilement travailler avec ces fonctions même si elles ont certaines limitations. PYTHON est le mieux adapté à toutes les opérations mathématiques et a également la capacité de relier des bibliothèques externes. PYTHON possède des fonctions de la boîte à outils de GA qui peuvent être combinées avec les fonctions de la boîte à outils EPANET. Afin de développer un modèle d'optimisation pour la conception du réseau de distribution d'eau.

Les techniques d'optimisation traditionnelles nécessitent des évaluations fonctionnelles et des temps de fonctionnement importants pour déterminer la solution optimale des problèmes du réseau de distribution par contre l'algorithme de Shuffled . Vu que les avantages de cette dernière . on le utilise pour notre cas dans le cadre de l'optimisation d'un système d'AEP .

Chapitre -I I -

**ALGORITHME
EVOLUTIONNAIRE**

II Introduction

De nombreux chercheurs ont développé de différentes approches pour une conception optimale des réseaux d'alimentation en eau. Mais aucun d'entre eux ne fournit de codage détaillé pour la procédure de conception concernée. Les étudiants et les jeunes chercheurs qui travaillent dans le domaine des réseaux de distribution d'eau passent généralement leur temps précieux à la recherche de la procédure pour relier l'algorithme génétique (GA) et EPANET. Par conséquent, afin de faciliter la tâche des jeunes chercheurs et étudiants, une procédure de conception détaillée utilisant le solveur EPANET avec des algorithmes génétiques dans l'environnement PYTHON pour une conception optimale basée sur la fiabilité des réseaux d'alimentation en eau

La méthodologie de l'AG est un processus évolutif, essentiellement imitant le processus d'évolution de la nature. GA est essentiellement une méthode de recherche efficace pour les cas d'optimisation non linéaire. Les opérations génétiques ont lieu au sein de la population des chromosomes. Par l'intermédiaire des divers opérateurs les connaissances génétiques des chromosomes changent continuellement et le succès de la population augmente progressivement à la suite de ces opérations. L'optimisation GA est également bien adaptée à l'optimisation des systèmes de distribution d'eau, en particulier les systèmes grands et complexes. L'objectif principal de cette étude est l'optimisation d'un réseau de distribution d'eau par GA. Les opérations d'AG sont réalisées sur un programme spécial qu'on va développer. Ce programme optimise les réseaux de distribution d'eau en tenant compte uniquement du coût en capital des conduites. [24]

Cinq opérateurs sont impliqués dans l'algorithme du programme. Ces opérateurs sont la génération, la sélection, l'élitisme, le crossover et la mutation. La taille optimale de la population se situe entre 30 et 70 selon la taille du réseau (c.-à-d. le nombre de tuyaux) et le nombre de tuyaux disponibles sur le marché. Taux d'élitisme devrait être d'environ 10 pour cent. Le taux de mutation devrait être sélectionné autour de 1-5 pour cent en fonction de la taille du réseau. Il est conseillé d'effectuer des croisements multipoints et des débits plus élevés. De plus, les paramètres de pénalité de pression sont très importants que les paramètres de vitesse. Le paramètre de pénalité inférieur à la pression est le plus important et devrait être environ 100 fois plus élevé que l'autre. On voit aussi dans ce chapitre une petite présentation sur logiciel de simulation Epanet et la langage de python.[25]

II.1 Qu'est-ce que l'algorithme génétique ?

Algorithme génétique est le membre d'une classe d'algorithmes de recherche basés sur l'évolution artificielle (Holland, 1975). L'algorithme génétique est la mise en œuvre de la théorie de l'évolution de Darwin dans les applications d'optimisation. Dans cette méthode, les variables sont présentées sous forme de nombres sur une chaîne appelée gènes et chromosomes respectivement. Avec l'aide de certaines opérations mathématiques, les chromosomes sont développés au cours des générations selon leur forme. L'évaluation de l'« aptitude » est fondée sur la mesure dans laquelle la solution d'essai répond à la « fonction de l'optimisation. Dans chaque génération, les chromosomes avec des meilleures valeurs de fitness (forme physique) survivent; d'autre part, les chromosomes les plus faibles sont éliminés en raison de leur faible fitness. La sélection naturelle garantit que les chromosomes ayant une meilleure aptitude se propageront dans les prochaines populations.[24]

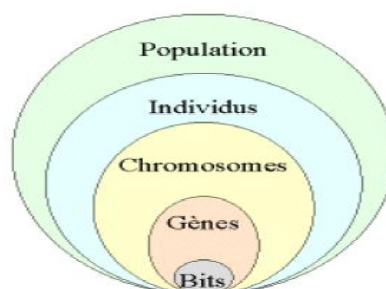


Figure II -1: Les cinq niveaux d'organisation d'un algorithme génétique (Goldberg, D.E 1989)[26]

II.1.1 Mécanisme de l'algorithme génétique

Les parties spécifiques de l'algorithme génétique qui ont une fonction spéciale sont appelées opérateurs. Dans sa forme la plus simple, un algorithme génétique se compose des trois opérateurs de base :

- Sélection
- Crossover
- Mutation

En plus de ces opérateurs de base, l'opérateur de génération crée la population initiale de chromosomes. De plus, l'opérateur d'élitisme est utilisé dans cette étude qui empêche la perte de chromosomes individuels réussis. Ces opérateurs sont appliqués à la génération actuelle pour former la génération suivante. L'algorithme génétique se poursuit jusqu'à ce que les critères de conception soient atteints et définis par l'utilisateur au début du projet. La figure II.1 présente les fondements de l'algorithme génétique. Au début, la population est évaluée et leurs aptitudes (condition physique) sont déterminées. Ensuite, les candidats retenus sont choisis et ils remplacent les candidats non retenus. La prochaine étape consiste à former la prochaine population en utilisant des opérateurs d'élitisme, de crossover et de mutation. Ces processus se poursuivent jusqu'à ce que le nombre de personnes prédéfini soit atteint. [24]

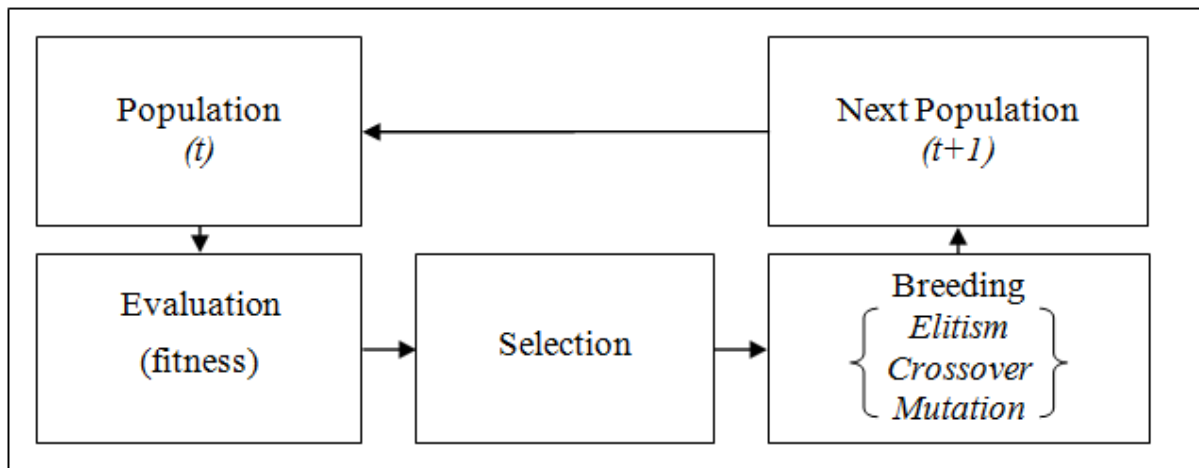


Figure II -2: Organigramme de l'algorithme génétique simple [24]

II.1.1.1 Concept des chromosomes

L'algorithme génétique utilise le concept des chromosomes pour définir les variables. Chaque variable de décision (comme la taille des tuyaux, le réglage des pompes, etc.) est définie dans les gènes pour former les chromosomes. La façon commune d'encoder est une chaîne binaire. Un exemple simple peut être formulé comme suit dans le tableau II.1. [24]

Tableau 0-1 : Exemple de codage binaire [24]

Taille de tuyau	Code binaire
100 mm	00
200 mm	01
300 mm	10
400 mm	11

II.1.1.2 Génération :

Sur la base des données attribuées au début du projet, l'algorithme génétique générera une population initiale de taille définie en utilisant un générateur de nombres aléatoires. Chaque population est composée de chromosomes. Chaque solution (chromosome) contiendra des variables de décisions générées au hasard. Le générateur de nombres aléatoires assigne, soit 1, soit 0 à chaque position de bit dans le chromosome où le nombre défini de bits représente des variables de décision spécifiques. Cet opérateur est appelé opérateur « Génération ». Par exemple, l'opérateur de génération produit une population comprenant trois

chromosomes illustrés ci-dessous à la figure II.2 avec trois chromosomes pour dix tuyaux en utilisant le codage ci-dessus.[26]

	tube 1		tube 2		tube 3		tube 4		tube 5		tube 6		tube 7		tube 8		tube 9		Pipe 10	
Chrom.1	1	1	0	1	1	1	0	0	0	1	0	1	1	0	0	0	0	1	1	1
	400		200		400		100		200		200		300		100		200		400	
Chrom.2	0	1	1	1	0	1	1	1	0	0	0	1	0	0	1	1	0	0	0	0
	200		400		200		400		100		200		100		400		100		100	
Chrom.3	1	0	0	0	1	0	1	1	0	1	1	0	1	0	1	0	1	1	0	1
	300		100		300		400		200		300		300		300		400		200	

Figure II -3 : Exemple de codage de tuyaux en binaire[26]

Une fois la population initiale générée, l’algorithme génétique traduira chaque gène en variable correspondante (c.-à-d. la taille du tuyau) et calculera la fonction objective (c.-à-d. le coût total). Une fois la fonction objective atteinte, une analyse sera effectuée pour chaque chromosome de la population et les lacunes de performance seront déterminées. Ces lacunes sont définies au stade de la mise en place du problème. Par exemple, l’obtention d’une pression acceptable aux nœuds dans un intervalle de pression défini ou l’obtention d’une vitesse acceptable aux tuyaux dans une plage de vitesse définie n’impliquera aucun coût total. D’autre part, l’algorithme génétique attribue un coût total à chaque solution (c.-à-d. dollars pour chaque perte de tête) qui ne répond pas aux critères prédéfinis de l’utilisateur. Le coût total est calculé en additionnant le coût de la canalisation et le coût total connexe. L’algorithme génétique calculera un niveau d’aptitude pour chaque solution dans la population en fonction d’une fonction du coût total de la solution. L’aptitude de chaque individu est déterminée en divisant sa valeur de pénalité par la valeur totale de la pénalité.[26]

II.1.1.3 Sélection :

Cet opérateur est utilisé pour éliminer les pires chromosomes en raison de leur faible condition physique. Une fois que leurs fonctions objectives sont déterminées à un stade précoce, un certain nombre des chromosomes présentant la pire aptitude est remplacé par le même nombre des meilleurs chromosomes.[26]

II.1.1.4 Élitisme :

L’élitisme est utilisé pour protéger les chromosomes les plus aptes contre les opérations de croisement et de mutation. L’objectif est d’avoir certains des meilleurs chromosomes comme ils le sont dans la prochaine génération et de ne pas les perdre. Élitisme peut rapidement augmenter la performance de l’algorithme génétique.[26]

II.1.1.5 Crossover :

La pression de sélection et la mutation aléatoire peuvent vous amener à une solution, mais ce n'est pas un AG (c'est ce qu'on appelle une "marche aléatoire"). La capacité d’échange de séquences entre différentes solutions pré-testées (encore vivantes) est essentielle à l’efficacité étonnante d’une AG dans la recherche dans de vastes espaces de résultats. Générer de la variété grâce à l’échange de séquence vous évite de perdre des groupes de bits utiles par pure malchance, mais leur donne plutôt la possibilité de se recombiner de nouvelles façons. Le terme biologique pour ce brassage sans danger est "Crossover". [26]

L’opérateur de crossover est appliqué afin d’initier un échange partiel des bits (informations) entre les chaînes mères pour former deux chaînes descendantes. L’algorithme génétique choisira au hasard deux solutions pour la reproduction. Le nombre total des croisements est défini par l’utilisateur au début de l’étude. La plupart des croisements sont des croisements à un point, à deux points et à plusieurs points, comme le montrent les figures ci-dessous. Notez que tous les points de croisement sont choisis au hasard.[26]

II.1.2 Mise en œuvre de l'algorithme génétique

Un exemple simple démontrant les deux premières étapes d'un algorithme génétique simple aidera à comprendre le processus. Les chiffres correspondants sont tirés de Goldberg (1989). Considérez le problème de maximiser la fonction $f(x) = x^2$, où x est autorisé à varier entre 0 et 31. Avec un entier non signé de cinq bits (chiffre binaire), nous pouvons obtenir des nombres entre 0 (00000) et 31 (11111). Nous simulons maintenant une seule génération d'algorithme génétique avec reproduction, croisement et mutation. Pour commencer, nous sélectionnons une population initiale. Nous sélectionnons une population de taille 4 en lançant une pièce juste 20 fois.[26]

Tableau 0-2: Illustration de l'algorithme génétique, ronde 1 [26]

Cordage Id	Population Initiales	x Valeur	f(x)	pselect _i	Nombre prévu	Nombre réel
	(Générée au hasard)	(Entier)	x^2	$f_i / \Sigma f$	f_i / f	
⊕ 1	0 1 1 0 1	13	169	0.14	0.58	1
2	1 1 0 0 0	24	576	0.49	1.97	2
3	0 1 0 0 0	8	64	0.06	0.22	0
4	1 0 0 1 1	19	361	0.31	1.23	1
Somme			1170	1.00	4.00	4
Moyenne			293	0.25	1.00	1
Maximum			576	0.49	1.97	2

Nous sélectionnons le pool d'accouplement de la prochaine génération en tournant la roulette pondérée quatre fois. La simulation réelle de ce processus au moyen de tirages au sort a fait en sorte que la chaîne 1 et la chaîne 4 ont reçu une copie dans le pool d'accouplement, la chaîne 2 a reçu deux copies, et la chaîne 3 n'a reçu aucune copie, comme le montre le tableau II.3 ci-dessus. En comparant le nombre réel de copies avec le nombre prévu de copies ($n \cdot pselect_i$), on obtient qu'il faut s'attendre à ce que les chromosomes les plus aptes obtiennent plus de copies, la moyenne reste égale, et le pire meurt.[26]

Tableau 0-3: Illustration de l'algorithme génétique, ronde 2[26]

Piscine d'accouplement après reproduction	Mate	Site de croisement	New Population	x Value	f(x)
(Site croisé montré)	(Au hasard)	(Au hasard)			x^2
0 1 1 0	2	4	0 1 1 0 0	12	144
1 1 0 0	1	4	1 1 0 0 1	25	625
1 1 0 0	4	2	1 1 0 1 1	27	729
1 0 1 1	3	2	1 0 0 0 0	16	256
Sum					1754
Average					439
Maximum					729

Ayant un pool de cordes, il est observé que le crossover simple se déroule en deux étapes :

- (1) les cordes sont accouplées au hasard, en utilisant des lancers de pièces pour jumeler les couples, et
- (2) Couples à cordes appariés croisés, en utilisant des lancers de pièces pour sélectionner les sites de croisement. Le croisement à un seul point est appliqué dans cet exemple. Pour revenir au tableau II.3, le choix aléatoire des partenaires a pris la deuxième chaîne dans le bassin d'accouplement à accoupler avec la première. Avec un site de croisement de 4, les deux cordes 01101 et 11000 traversent et produisent deux nouvelles

cordes 01100 et 11001. Les deux autres cordes de la piscine d'accouplement sont croisées au site 2.

Le dernier opérateur, la mutation est effectuée petit à petit. Nous supposons que la probabilité de mutation dans ce test 0,001. Avec 20 bits transférés, nous devrions nous attendre à ce que $20 \times 0,001 = 0,02$ bits subissent une mutation au cours d'une génération donnée. La simulation de ce processus indique qu'aucun bit ne subit de mutation pour cette valeur de probabilité.

Après reproduction, croisement et mutation, la nouvelle population est prête à être testée. Pour ce fait, nous décodons simplement les nouvelles chaînes créées par l'algorithme génétique simple et calculons les valeurs de fonction de fitness à partir des valeurs x ainsi décodées. La condition physique moyenne de la population est passée de 293 à 439 en une génération. L'aptitude maximale est passée de 576 à 729 au cours de la même période.

Bien que les processus aléatoires aident à provoquer ces circonstances heureuses, nous commençons à remarquer que cette amélioration n'est pas une coïncidence. La meilleure chaîne de la première génération (11000) reçoit deux copies en raison de ses performances élevées et supérieures à la moyenne. Lorsque cela se combine au hasard avec la chaîne la plus haute suivante (10011) et est croisée à l'emplacement 2 (encore une fois au hasard), l'une des chaînes résultantes (11011) s'avère être un très bon choix en effet.

II.1.3 Domaines d'application :

Les algorithmes génétiques sont de plus en plus utilisés. On les retrouve dans les domaines suivants : [27]

- Optimisation : optimisation de fonctions, planifications, etc ...
- Apprentissage : classification, prédiction, robotique, etc ...
- Programmation automatique : programmes LISP, automates cellulaires, etc ...
- Etude du vivant, du monde réel : marchés économiques, comportements sociaux, systèmes immunitaires, etc ...

II.1.4 Algorithmes génétiques dans le domaine de l'eau :

Les algorithmes génétiques dans le domaine de l'alimentation en eau potable (AEP) concernent les domaines suivants : [27]

- Utilisation pour le dimensionnement des réseaux d'AEP ;
- Utilisation pour la réhabilitation des réseaux d'AEP ;
- Recherche des fuites dans les réseaux d'AEP ;
- Optimisation des emplacements des stations de chloration dans les systèmes d'AEP;
- Détection des sources de pollution dans les réseaux d'AEP ;
- Optimisation du fonctionnement des stations de pompage ;
- Calage automatique des modèles hydrauliques des réseaux AEP.
- Choix optimal d'une extension.

II.2 couplage SCE et EPANET :

Dans notre étude on va coupler une technique d'optimisation, Shuffled Evolution Complex (SCE), et un modèle de simulation, EPANET.

La conception d'un réseau de canalisations fiable et économique pour les réseaux de distribution d'eau est d'une grande importance. L'étude vise à mettre en place un schéma de conception et de réhabilitation du réseau d'eau. Le modèle se compose d'un algorithme d'optimisation et d'un solveur réseau. Un algorithme d'optimisation efficace, Shuffled Complex Evolution (SCE), ainsi qu'un solveur réseau EPANET sont utilisés pour arriver au schéma proposé. Le SCE est choisi dans cette étude parce qu'il recherche dans toutes les directions en se basant sur les techniques de recherches simplex de Nelder et Mead, ce qui a peu de chance d'être piégé dans les optima locaux. EPANET a été choisi parce qu'il permet de simuler les réseaux de distribution d'eau en régime permanent et en période prolongée. En plus, cet outil peut également être utilisé pour la modélisation de la qualité de l'eau. Ce chapitre présente des discussions sur le modèle de simulation EPANET, le mécanisme de fonctionnement global de la technique de recherche locale ShuffledComplex, Nelder et Mead, la sélection des paramètres algorithmiques SCE, les critères d'arrêt pour l'algorithme, la capacité d'exploration et d'exploitation. Le schéma proposé et la procédure de liaison entre la

SCE et EPANET sont décrits ci-après. Enfin, le chapitre se termine par des remarques finales générales.[28]

II.2.1 Ce qu'est EPANET

EpaNet est un programme informatique en langage C qui permet de simuler pendant une période prolongée le comportement hydraulique et la qualité de l'eau dans les réseaux de canalisations sous pression. Un réseau se compose de tuyaux, des nœuds (jonctions de tuyaux), des pompes, des vannes et des réservoirs de stockage. EpaNet suit le débit d'eau dans chaque conduite, la pression à chaque nœud, la hauteur d'eau dans chaque réservoir et la concentration d'espèces chimiques dans l'ensemble du réseau pendant une période de simulation composée de plusieurs étapes temporelles. En plus des espèces chimiques, on peut aussi simuler l'âge de l'eau et le traçage des sources.[29]

EpaNet est conçu comme un outil de recherche pour améliorer notre compréhension du mouvement et du devenir des constituants de l'eau potable dans les réseaux de distribution. Il peut être utilisé pour différents types d'applications dans l'analyse des systèmes de distribution. La conception du programme d'échantillonnage, l'étalonnage du modèle hydraulique, l'analyse du chlore résiduel et l'évaluation de l'exposition des consommateurs en sont quelques exemples. EpaNet peut aider à évaluer d'autres stratégies de gestion pour améliorer la qualité de l'eau dans l'ensemble d'un système. Il peut s'agir notamment[29]

- * La modification de l'utilisation des sources dans les systèmes à sources multiples,
- * Modifier le régime de pompage ou de marnage des réservoirs,
- * Préciser l'usage des stations de chloration (ou autres retraitements) en réseau,
- * Planifier l'entretien et le remplacement de certaines canalisations.

L'édition des données de réseau, pour l'exécution de simulations hydrauliques et des simulations qualité, et pour l'affichage des résultats sous plusieurs formats (descartes avec des codes couleurs, des tableaux et des graphiques).

II.2.2 Capacité de modélisation

L'interface utilisateur d'EPANET fournit un éditeur de réseau visuel qui simplifie le processus de création de modèles de réseau de canalisations et la modification de leurs propriétés et de leurs données. Divers outils de création de rapports et de visualisation de données sont utilisés pour interpréter les résultats d'une analyse de réseau, notamment des cartes de réseau codées par couleur, des tableaux de données, la consommation d'énergie, la réaction, l'étalonnage, des graphiques de séries chronologiques ainsi que des courbes de profil et de contour.[30]

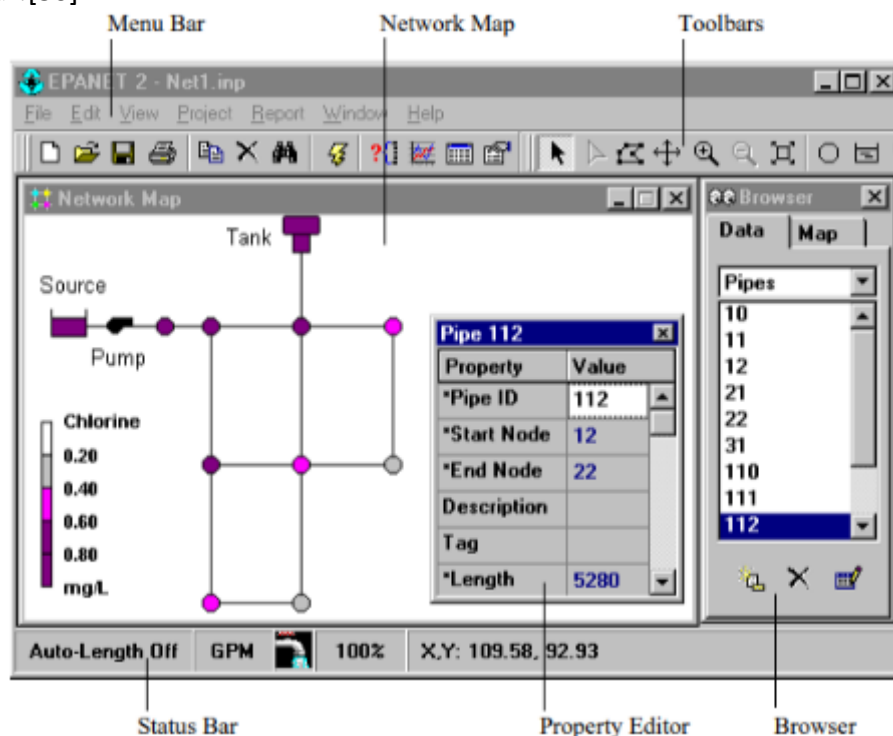


Figure 0-6: Espace de travail EpaNet [30]

EPANET applique une approche coordonnée pour modéliser à la fois l'hydraulique des réseaux et la qualité de l'eau. Les équations hydrauliques de base impliquées dans EPANET sont brièvement décrites ci-dessous :

1. Les équations d'écoulement dans le modèle hydraulique sont régies par la conservation de la masse et de l'énergie. La loi de la conservation de masse (loi des nœuds) stipule que le taux de stockage dans un système est égal à la différence entre le flux entrant et le flux sortant vers le système.

Pour chaque jonction, la conservation de la masse peut s'exprimer comme suit :[30]

$$\sum Q_{entr} - \sum Q_{sort} = Q_{ext} \dots\dots\dots (\text{II.1})$$

Où Q_{entr} et Q_{sort} sont les entrées et les sorties du nœud ;
Et Q_{ext} est la demande externe (débit au nœud) ;

2. La conservation de l'énergie indique que le long d'un parcours orienté et fermé (boucle), la somme algébrique des pertes de charge doit être nulle. En pratique, cette condition est rarement satisfaite d'une façon rigoureuse le long de toute boucle fermée on tolère une différence de 0.5m, la perte totale peut s'écrire comme suit :

$$\sum_{i \in Ip} h_{ji} - \sum_{Mx \in Jp} PE = 0 \dots\dots\dots (\text{II.2})$$

Où $h_{i,j}$ est la perte de charge dans les noeuds de raccordement i et j ;
 Ip est l'ensemble des tuyaux dans la boucle P ;
 Mx désigne les pompes ;
 Jp est l'ensemble des pompes dans la boucle P ;
 PE est l'énergie ajoutée par la pompe Mx .

3. La perte de charge dans le tube ou charge hydraulique perdue à cause du frottement de l'eau avec les parois du tuyau peut être calculée en utilisant une de ces trois formules :
 - formule de Hazen-Williams
 - formule de Darcy-Weisbach
 - formule de Chezy-Manning

La formule de Hazen-Williams est la formule de perte de charge la plus utilisée aux États-Unis. Elle ne peut pas être utilisée pour des liquides autres que l'eau et a été initialement développée uniquement pour les écoulements turbulents. La formule de Darcy-Weisbach est théoriquement la plus correcte et est la plus utilisée en Europe. Elle s'applique à tous les régimes d'écoulement et à tous les liquides. La formule de Chezy-Manning est généralement utilisée pour les écoulements dans les canaux découverts et pour les grands diamètres. La formule de Darcy-Weisbach est sélectionnée par défaut. Chaque formule utilise l'équation suivante pour calculer la perte de charge linéaire entre les nœuds de début et de fin du tuyau:[29]

$$h_L = Aq^B \dots\dots\dots (\text{II.3})$$

dans laquelle h_L est la perte de charge (en unités de longueur),
 q : Le débit (Volume/Temps),
 A : Le coefficient de résistance,
 B : L'exposant du débit.

Pour la formule de Darcy-Weisbach, EPANET utilise des différentes méthodes pour calculer le coefficient de frottement linéaire λ :

- La formule de Nikuradzé :

$$\lambda = (1.14 - 0.86 \ln \frac{Ru}{D})^{-2} \dots\dots\dots (\text{II.4})$$

- La formule de Colebrook :

$$\lambda = [-0.86 \ln (\frac{Ru}{3.7D} + \frac{2.51}{Re \sqrt{\lambda}})]^{-2} \dots\dots\dots (\text{II.5})$$

Ru : désigne la rugosité absolue moyenne de la paroi interne de la conduite.

Re : nombre de Reynolds.

D : diamètre interne de la conduite.

II.2.2.1 Capacité pour modélisation hydraulique :

Une modélisation hydraulique complète et précise est une condition préalable à la modélisation efficace de la qualité de l'eau. EPANET contient un moteur d'analyse hydraulique de pointe qui inclut les fonctionnalités suivantes:[30]

- Fonctionnement du système basé à la fois sur des contrôles simples de niveau de réservoir ou de minuterie et sur des contrôles complexes basés sur des règles.
- Aucune limite sur la taille du réseau pouvant être analysée.
- Calcule la perte de charge à l'aide des formules Hazen-Williams, Darcy-Weisbach ou Chezy-Manning.
- Comprend les pertes de charge mineures pour les coudes, les raccords, etc.
- Modèles de pompes à vitesse constante ou variable.
- Calcule l'énergie de pompage et coûte 10.
- Modèles différents types de vannes, y compris les vannes d'arrêt, de contrôle, de régulation de pression et de débit.
- Permet aux réservoirs de stockage d'avoir n'importe quelle forme (le diamètre peut varier avec la hauteur).
- Considère plusieurs catégories de demandes au niveau des nœuds, chacune avec son propre modèle de variation temporelle.
- Modèles de débit en fonction de la pression provenant d'émetteurs (têtes de sprinklers).

II.2.2.2 Capacités pour la Modélisation de la Qualité de l'Eau :

En plus des simulations hydrauliques, EPANET peut modéliser la qualité d'eau; on dispose ainsi des capacités suivantes: [30]

- Modélisation du déplacement d'un traceur pendant la durée de la simulation.
- Modélisation des déplacements et les variations de concentration en plus ou en moins que subit une substance (par exemple un produit secondaire de désinfection, ou du chlore résiduel).
- Modélisation du temps de séjour de l'eau dans le réseau.
- Indication à chaque n.ud de la proportion d'eau provenant d'une ressource distincte.
- Modélisation des réactions d'évolution de la qualité de l'eau dans la masse et aux parois.
- Utilisation des cinétiques d'ordre n pour calculer les réactions dans la masse d'eau.
- Utilisation des cinétiques d'ordre un ou zéro pour calculer les réactions aux parois des tuyaux.
- Limitations de transfert de masse pour modéliser les réactions aux parois.
- Fixation d'une concentration limite des réactions.
- Utilisation des coefficients de vitesse de réaction globale et des coefficients spécifiques pour certains tuyaux.
- Etablissement d'une corrélation entre les coefficients de vitesse de réaction au niveau de la paroi en fonction de la rugosité du tuyau.
- Introduction d'une substance quelconque à différents emplacements du réseau variant dans le temps en débit massique ou en concentration.
- Mélange d'eau dans les réservoirs : mélange parfait, à flux piston ou en deux compartiments.

En utilisant ces capacités, EPANET peut modéliser des phénomènes en rapport avec la qualité de l'eau comme:[30]

- Le mélange d'eau provenant de différentes sources;
- Le temps de séjour de l'eau dans le réseau;
- La diminution du chlore résiduel;
- L'accroissement des sous-produits de la désinfection;
- La diffusion d'un polluant dans le réseau, introduit en certains points.

II.2.3 Étapes de l'utilisation d'EpaNet :

En règle générale, on exécute les étapes suivantes lorsqu'on utilise EpaNet pour modéliser un système de distribution d'eau :[30]

1. Dessiner une représentation réseau de votre système de distribution ou importer une description de base du réseau placée dans un fichier texte.
2. Saisir les propriétés des objets qui composent le système.
3. Décrire le fonctionnement du système.
4. Sélectionner un ensemble d'options de simulation.
5. Lancer une simulation de la qualité de l'eau/hydraulique
6. Visualiser les résultats d'une simulation.

II.2.4 Modèle de simulation hydraulique :

EpaNet est un modèle de simulation hydraulique qui calcule les têtes de jonction et les débits de liaison pour un ensemble fixe de niveaux de réservoir et de demandes en eau sur une série de points dans le temps. D'une étape de temps à l'autre, les niveaux des réservoirs et les demandes de jonction sont mis à jour selon les schémas temporels prescrits, tandis que les niveaux des réservoirs sont mis à jour à l'aide de la solution de débit actuelle. La solution pour les hauteurs de chute et les débits à un moment donné consiste à résoudre simultanément la conservation de l'équation de débit pour chaque jonction et la relation de perte de charge à travers chaque liaison du réseau. Ce processus, appelé " équilibrage hydraulique " du réseau, nécessite l'utilisation d'une technique itérative pour résoudre les équations non linéaires impliquées. EpaNet utilise à cet effet l'algorithme de gradient". Le pas de temps hydraulique utilisé pour la simulation de période prolongée (EPS) peut être réglé par l'utilisateur. Une valeur typique est de 1 heure. Des pas de temps plus courts que la normale se produiront automatiquement chaque fois que l'un des événements suivants se produira :[30]

- La période de déclaration suivante se produit ;
- La période de modèle de temps suivante se produit ;
- Un réservoir devient vide ou plein ;
- Un contrôle simple ou un contrôle basé sur des règles est activé.

II.2.5 Algorithmes d'analyse hydraulique :

La méthode utilisée dans EpaNet pour résoudre les équations de continuité d'écoulement et de perte de charge qui caractérisent l'état hydraulique du réseau de canalisations à un moment donné peut être qualifiée d'approche hybride en boucle. Des informations détaillées sur la mise en œuvre de la solution hydraulique par EpaNet et d'autres informations sont disponibles dans le manuel EpaNet.[30]

II.2.6 SHUFFLED COMPLEX EVOLUTION (SCE):

Shuffled Complex Evolution (SCE) est un outil d'optimisation global développé à l'Université de l'Arizona (Duan et al, 1992). Le SCE a été utilisé pour divers problèmes d'ingénierie par différents chercheurs et il a été démontré que cet algorithme est efficace et efficient pour une vaste gamme de problèmes (Duan et al, 1992 ; Duan et al, 1993).[31]

SCE (Thyer et al, 1999) travaille sur la base de quatre concepts :

- (1) Combinaison d'approches déterministes et probabilistes.
- (2) Evolution systématique d'un ensemble de points.
- (3) Evolution concurrentielle.
- (4) Mélange complexe.

L'algorithme commence par une population des points sélectionnés au hasard dans l'espace réalisable à l'aide d'un générateur des nombres pseudo-aléatoires. L'échantillon des points est limité par les limites inférieures et supérieures des valeurs des paramètres. Cette génération aléatoire des points permet de localiser l'optimum global sans être biaisée par des points prédéfinis. Les points sont triés par ordre croissant de valeur de critère de sorte que le premier point représente la plus petite valeur de fonction et que le dernier point représente la plus grande valeur de fonction. La population initiale générée aléatoirement est divisée en plusieurs complexes de $(2N + 1)$ points chacun, où N est la dimension du problème d'optimisation. Chaque complexe est autorisé à évoluer indépendamment pour rechercher le domaine réalisable dans des directions différentes. Chaque point individuel d'un complexe a le potentiel de participer au processus de reproduction de nouveaux points. De chaque

complexe, quelques points (N + 1) sont sélectionnés pour former un sous-complexe, où le Nelder modifié et le Mead Simplex[31]

(NMSM) est appliquée pour l'amélioration globale. Les points d'une bonne condition physique ont plus de chances d'être sélectionnés pour engendrer une progéniture. Le NMSM effectue des étapes de réflexion et de contraction interne pour obtenir un meilleur point d'ajustement. Cette nouvelle progéniture remplace le pire point du simplexe. Les points dans les complexes évolués sont combinés en une population échantillon. La population de l'échantillon est à nouveau triée, mélangée et réassignée dans des nouveaux complexes pour permettre le partage de l'information. Ce processus est répété jusqu'à ce que certains critères d'arrêt soient satisfaits.[31]

Une description détaillée des étapes de l'algorithme d'optimisation SCE (Duan et al, 1994) est donnée ci-dessous et illustrée à la figure II.7:

1. Initialiser les complexes des nombres (p) et le nombre des points dans chaque complexe (m). Calculer la taille de l'échantillon $S = p \times m$.
2. Générer S population des points au hasard dans l'espace de solution ($\Omega \subset \mathcal{R}^n$) en utilisant une distribution de probabilité uniforme. Chacune des populations S représente une combinaison possible de N paramètres (x_1, x_2, \dots, x_n).
3. Calculer la valeur du critère (f_i) aux solutions (x_i).
4. Classer les points : trier les points S dans l'ordre d'augmentation de la valeur de fonction et stocker les dans un tableau $D = \{x_i, f_i, i=1, \dots, S\}$, de sorte que le premier point ($i = 1$) représente la plus petite valeur de critère et le dernier point ($i = S$) représente la plus grande valeur de critère.
5. Partition en complexes (Figure II.9) - Partitionner le D en p complexes (A_1, \dots, A_p), chacun contenant m points. Les complexes sont divisés de telle sorte que le premier complexe contienne chaque p (j-1)+1 point classé, le second complexe contient chaque p (j-1)+2 point classé, et ainsi de suite, où $j = 1, 2, \dots, m$. Cela peut s'exprimer sous forme mathématique :

$$A^k = \{x_j^k, f_j^k \mid x_j^k = x_{k+p(j-1)}, f_j^k = f_{k+p(j-1)}\} \dots \dots \dots (II.6)$$
6. Evoluer chaque complexe : faire évoluer chaque complexe ($A^k, k = 1, \dots, p$) selon l'algorithme d'évolution du complexe compétitif (CCE) qui sera discuté dans la section suivante.
7. Shuffle complexes : Combiner les points des complexes évolués en une seule population d'échantillons (remplacer A_1, \dots, A_p par D) ; trier la population de l'échantillon dans l'ordre d'augmentation de la valeur du critère ; Shuffle (c.-à-d. redistribuer) la population de l'échantillon en p complexes selon la procédure spécifiée à l'étape 5.
8. Vérifier la convergence : Si l'un des critères de convergence prédéfini est satisfait, arrêter ; sinon continuer.
9. Vérifier la réduction du nombre des complexes - Si le nombre minimum des complexes requis dans la population, p_{min} , est inférieur à p, retirer le complexe ayant les points les moins bien classés; fixer $p = p - 1$ et $S = p \times m$; retourner à l'étape 6. Si $p_{min} = p$, revenez à l'étape 6.

II.2.6.1 Évolution du complexe compétitif (CCE)

Les algorithmes CCE (Figure II.8), basés sur le schéma de recherche en descente simplexe de Nelder et Mead (1965), sont présentés ci-dessous : [32]

1. Initialiser le nombre des points dans un sous-complexe (q), le nombre des descendants consécutifs générés par un sous-complexe (α), le nombre d'étapes d'évolution prises par chaque complexe (β), où $2 \leq q \leq m, \alpha \geq 1$, et $\beta \geq 1$.
2. Attribuer des poids - Attribuer une distribution de probabilités triangulaire à A^k ; c.-à-d,

$$p_i = \frac{2(m+1-i)}{m(m+1)} ; i = 1, \dots, m \dots \dots \dots (II.7)$$

Les points x_1^k ont la probabilité la plus élevée, $p_1 = \frac{2}{(m+1)}$, Le point x_m^k a la probabilité la plus faible $p_m = \frac{2}{m(m+1)}$

3. Construire un sous-complexe (figure II.9) en sélectionnant au hasard q points (u_1, \dots, u_q) du complexe (A^k) selon une distribution de probabilités triangulaire. La

distribution de probabilité est spécifiée de telle sorte que le meilleur point (c.-à-d. le point ayant la meilleure valeur fonctionnelle) a le plus de chance d'être choisi pour former le sous-complexe, et le pire point a le moins de chance. Les stocker dans le tableau $B = \{u_i, v_i, i = 1, \dots, q\}$, où v_i est la valeur de fonction associée au point u_i . Stocker en L, l'emplacement de A^k qui sont utilisés pour construire B.

4. Trier B et L de telle sorte que q points soient disposés dans l'ordre croissant de la valeur de fonction. Identifier le pire point du sous-complexe et calculer le centroïde (g) du sous-complexe sans inclure le pire point en utilisant l'expression :

$$g = \frac{1}{q-1} \sum_{i=1}^{q-1} U_i \dots\dots\dots (II.8)$$

Essayer une étape de réflexion (Figure II.10) en réfléchissant le point, le plus défavorable, à travers le centroïde. Si le point nouvellement généré ($r = 2g - u_q$) se trouve dans l'espace faisable (Ω), calculer la valeur de fonction (f_r) et passer à l'étape 6 ; sinon, générer aléatoirement un point (z) dans l'espace faisable et calculer f_z , ensemble $r = z$ et $f_r = f_z$.

5. Si le point nouvellement généré est meilleur que le point le plus mauvais, c'est-à-dire $f_r < f_q$, remplacer le point le plus mauvais (u_q) par le nouveau point (r). Passer à l'étape 9. Sinon, passer à l'étape 7.
6. Essayer une étape de contraction (Figure II.9) en calculant un point (c) à mi-chemin entre le centroïde (g) et le pire point (u_q), c'est-à-dire $c = \frac{(g + u_q)}{2}$ et la valeur de fonction f_c . Si le point de contraction est meilleur que le point le plus mauvais, c.-à-d. $f_c < f_q$, remplacer le point le plus mauvais (u_q) par le point de contraction (c) et passer à l'étape 9. Sinon, passer à l'étape 8.
7. Générer aléatoirement un point (z) dans l'espace réalisable et calculer f_z . Remplacer le point le plus défavorable (u_q) par le point généré aléatoirement (z).
8. Répéter l'étape 4 - Étape 8 α fois, où $\alpha \geq 1$ est le nombre des progénitures successives générées par chaque sous-complexe.
9. Répéter l'étape 1 - Étape 9 β fois, où $\beta \geq 1$ est le nombre d'étapes de l'évolution par chaque complexe.

II.2.6.2 Sélection des paramètres dans la méthode SCE :

Le nombre des complexes (p) et le nombre des points dans chaque complexe (m) sont les deux paramètres principaux de la SCE, puisqu'ils déterminent la taille totale de la population ($S = p \times m$). En général, plus la taille de la population (S) est grande, plus la probabilité de trouver des minima globaux, est élevée. L'efficacité de l'algorithme dépend de la plus grande valeur de S. Cependant, à mesure que S augmente, l'évaluation de la fonction objective augmente et l'efficacité de l'algorithme est affectée. Pour un petit problème, $S = N + 1$ devrait être suffisant pour trouver la solution optimale, tandis que pour un problème compliqué S sera plus grande.[32]

Le nombre des points dans un complexe (m) ne doit pas être trop élevé. Si la valeur m est trop grande, un temps de calcul excessif sera nécessaire. Il en résulte une réduction de l'efficacité des calculs. Duan et ses collaborateurs (1993) ont suggéré qu'une bonne valeur pour m est $(2N + 1)$ ou plus, où N est le nombre des paramètres à optimiser.

Le nombre des points dans chaque sous-complexe (q) peut varier entre deux et m. Cependant, une valeur de $(N + 1)$ est suggérée comme un bon choix, puisque $(N + 1)$ points définissent une approximation du premier ordre de la surface fonctionnelle.[32]

Le nombre de progénitures successives générées par chaque sous-complexe, α , peut être supérieur ou égal à un. Si α est égal à un, cela signifie qu'un seul des points sera remplacé avant que le sous-complexe soit replacé dans le complexe. Le nombre de sous-complexes choisis dans chaque complexe pour la reproduction, β , peut prendre un nombre supérieur à un. Si β est petit, les complexes seront fréquemment mélangés et la recherche ne parviendra pas à effectuer une exploration indépendante dans l'espace des paramètres. D'autre part, si β est grand, le complexe se déplacera vers des minima locaux. Cependant, $\beta = m$ s'est avéré être un bon choix.[32]

II.2.6.3 Critères d'arrêt de l'algorithme SCE :

Dans SCE, quatre critères d'arrêt sont vérifiés à chaque génération. Si l'un d'eux est satisfait, le calcul est immédiatement terminé. Les critères sont les suivants :[33]

1. Nombre maximal d'évaluations de fonctions : si la recherche atteint l'évaluation maximale de la fonction objective sans générer un point qui a une valeur de critère inférieure à une limite prédéfinie, les algorithmes se termineront. Ce critère est très utile dans les cas où le modèle prend du temps et doit s'arrêter au milieu d'une boucle.
2. Changement relatif de fonction d'objectif : si le changement relatif des valeurs de fonction d'objectif dans les k dernières boucles de brassage ne change pas plus qu'un pourcentage prédéfini, la recherche s'arrête. Ce critère implique que la "convergence" peut avoir été atteinte ou que le calcul a été pris au piège dans certains Optima locaux.
3. Valeur de la meilleure fonction : si la valeur de la meilleure fonction de l'objectif est inférieure à une certaine valeur, on considère que la valeur " optimale " a été atteinte.
4. Convergence de la population : si la population a convergé vers la valeur pré-spécifiée (p. ex., 0,001) de l'espace de paramètre original, les algorithmes se termineront.

II.2.6.4 Exploration et exploitation en SEC :

Exploration et exploitation sont les deux techniques utilisées par tout algorithme efficace dans la recherche d'une solution optimale. L'exploration explore des zones nouvelles et inconnues dans l'espace de recherche et l'exploitation utilise les informations importantes trouvées aux points précédemment explorés (Muttill et Liang, 2002). SCE offre un assez bon équilibre entre l'exploration et l'exploitation. SCE partage la population totale des points dans plusieurs communautés. La répartition de la population facilite une exploration approfondie de l'espace de solution dans différentes directions, réduisant ainsi la recherche se coincer dans Optima local. Il évolue à plusieurs reprises complexes basés sur Nelder et Mead simplex technique de recherche en évolution complexe compétitive et dirige ainsi la recherche dans la région réalisable. la population des complexes évolués est mélangée pour assurer le partage de l'information. Ces multiples changements complexes et l'évolution complexe assurent un équilibre efficace entre l'exploration et l'exploitation.[33]

II.2.6.5 Avantages du SCE par rapport à l'algorithme de recherche traditionnel :

Le SCE diffère à bien des égards des algorithmes de recherche traditionnels. Les différences les plus importantes sont les suivantes :[32]

1. La recherche SCE utilise une population des points en parallèle, et donc pas seulement un seul point. Le principal avantage de cette technique de recherche parallèle par rapport à la méthode traditionnelle, telles que le calcul et la recherche aléatoire, est qu'elle recherche dans toutes les directions simultanément. Les chances de converger vers un optimum local sont considérablement réduites.
2. La recherche SCE est basée sur la valeur de la fonction objective et ne dépend pas de l'information dérivée. Cependant, certaines méthodes traditionnelles comme les algorithmes de recherche par gradient nécessitent la construction ou l'approximation de l'information dérivée pour rechercher l'espace possible. Ces méthodes ne réussissent pas à localiser l'optimum global lorsque la surface de réponse présente des discontinuités.
3. Le SCE suit les règles probabilistes au lieu des règles déterministes pour passer d'une génération à l'autre. Dans le processus de reproduction, les solutions mères sont sélectionnées pour générer la progéniture en utilisant des règles de transition probabilistes. Les individus qui ont une valeur de fonction objective inférieure ont plus des chances d'être sélectionnés pour la procréation que ceux qui ont des valeurs de fonction élevées.
4. La recherche SCE maintient un équilibre supérieur entre l'exploration et l'exploitation.
5. La SCE arrondit immédiatement la valeur de décision réelle à la taille du marché la plus proche dans la conception du réseau de distribution d'eau et élimine les complexités qui se sont produites lors de l'arrondissement dans les méthodes traditionnelles (LP, NLP et DP). Il traite toujours des variables discrètes assurant la qualité de la solution.

II.2.7 SCHEMA PROPOSE

Dans la présente étude, SCE (Duan et al., 1992) est couplé à EPANET pour rechercher les variables décisionnelles optimales (p. ex., diamètres des conduites) dans des cas comme la conception des nouveaux réseaux ou la réhabilitation des réseaux existants. SCE produit des combinaisons possibles des solutions d'essai basées sur des approches probabilistes et déterministes et EPANET analyse le réseau à l'aide de ces valeurs variables proposées. Par la suite, SCE utilise le feedback d'EPANET (le retour d'expérience d'EPANET) pour améliorer la qualité des solutions. Pour appliquer le modèle, SCE est modifié pour prendre en charge des variables de décision avec un nombre plus élevé

II.2.7.1 Liaison d'EPANET avec SCE

Dans le schéma proposé, le solveur de réseau EPANET (code C) est lié à l'algorithme d'optimisation SCE (code PYTHON). Il existe un sous-programme de fonction appelé par le moteur principal de SCE pour évaluer les valeurs de la fonction objective. Une commande en langage « PYTHON » appelle le solveur réseau Epanet pour déterminer les performances hydrauliques du réseau. Ainsi, les variables de contrôle (diamètre du tuyau, coefficients de rugosité et demandes nodales) sont transmises de l'optimiseur SCE au solveur réseau par un fichier d'entrée formaté (.INP) d'EPANET. Le modèle de simulation résout les équations hydrauliques et détermine les valeurs de la tête de pression nodale, les débits de conduite, etc. La tête nodale résultant du modèle de simulation est renvoyée à l'optimiseur SCE. La stratégie est illustrée à la figure II.11. [25]

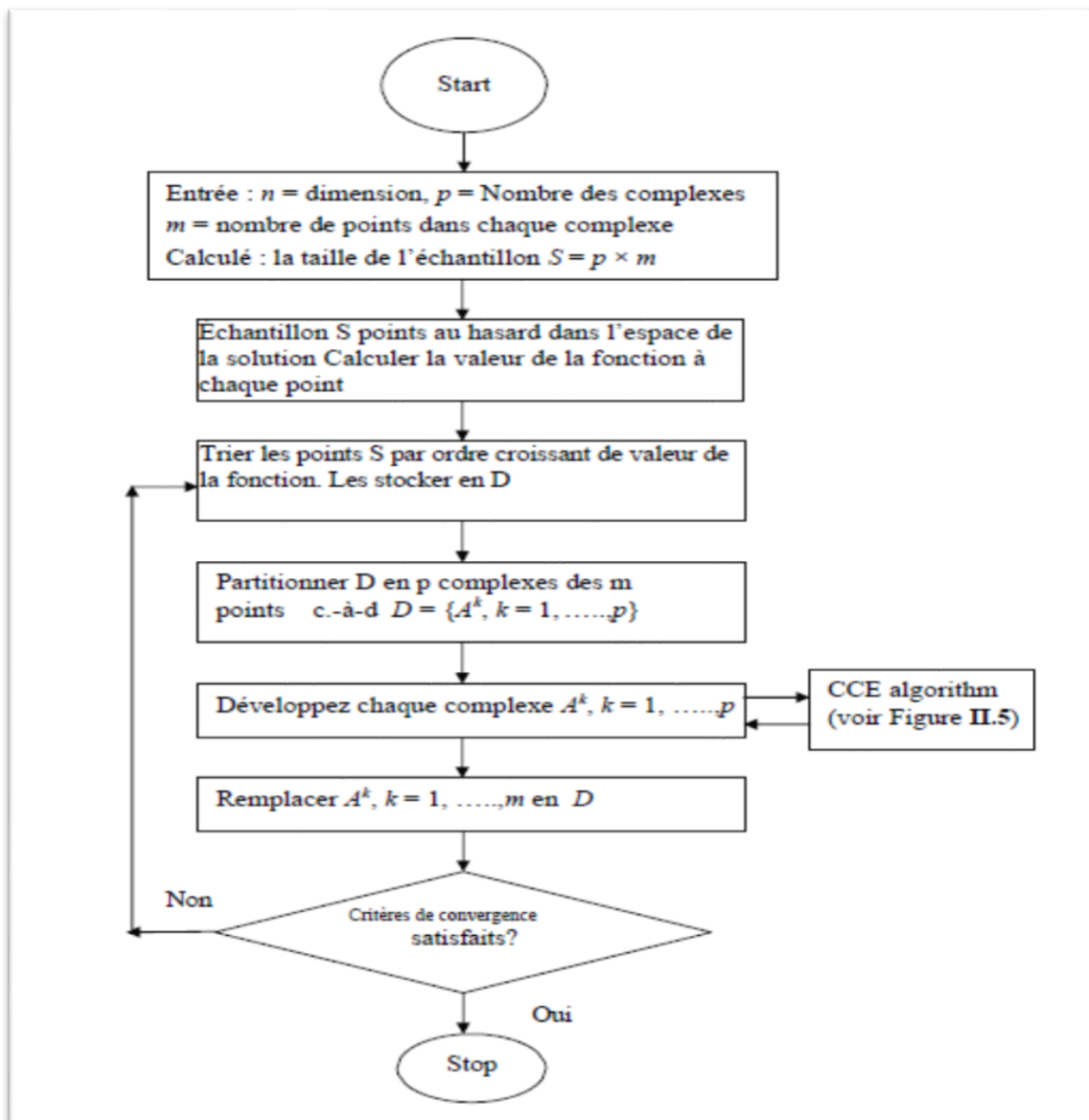


Figure II -7 : Organigramme de l'algorithme SCE [32]

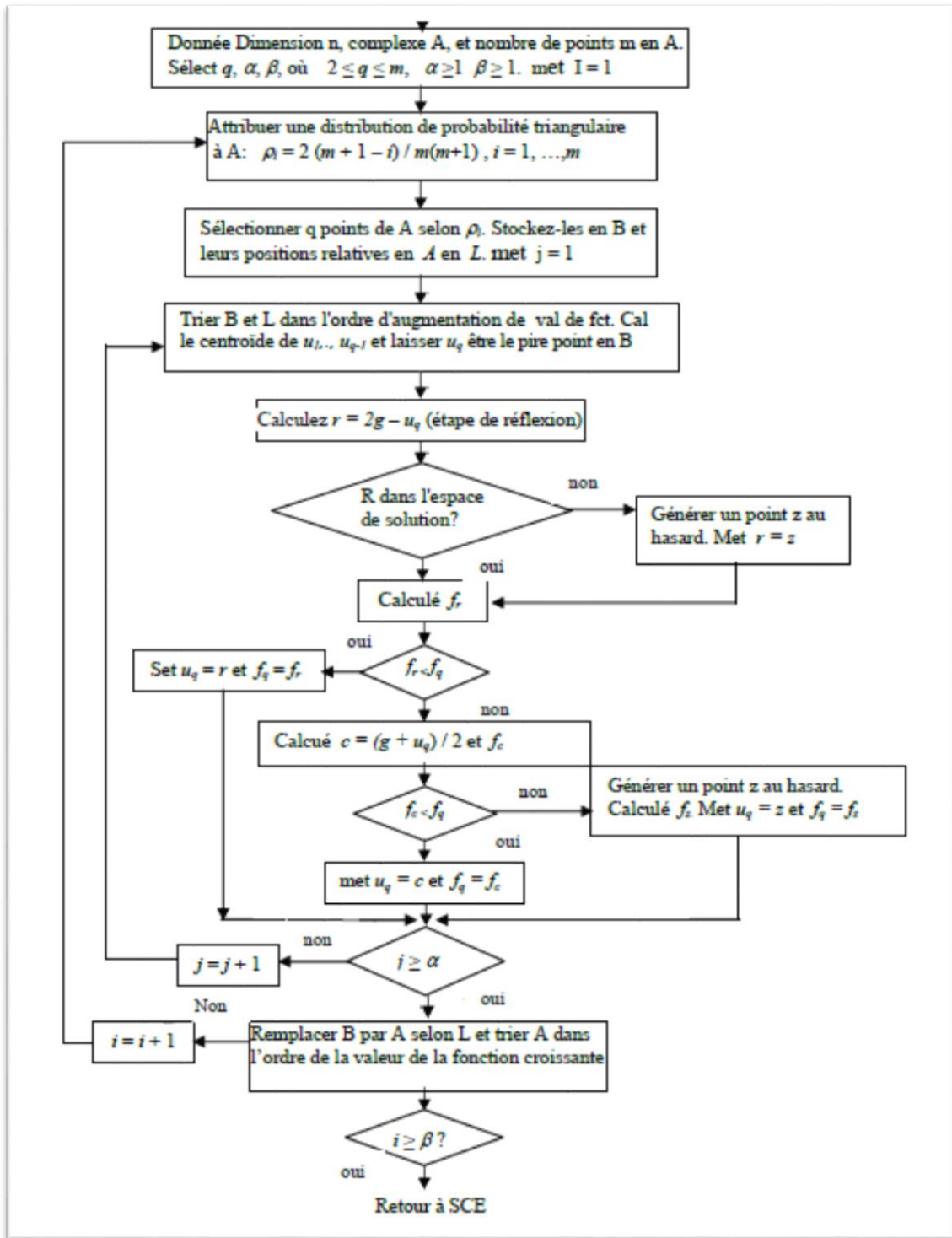


Figure II -8 : Organigramme de la stratégie CCE de l'algorithme SCE [32]

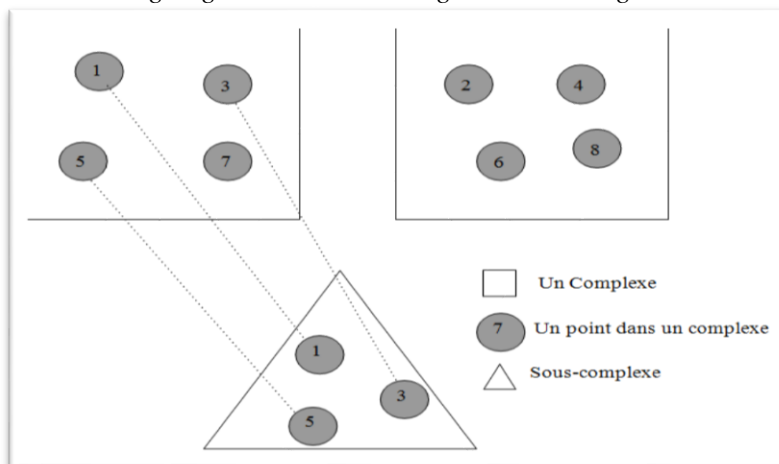


Figure II -9 : Évolution du complexe de Shuffled : Complexes et sous-complexes [32]

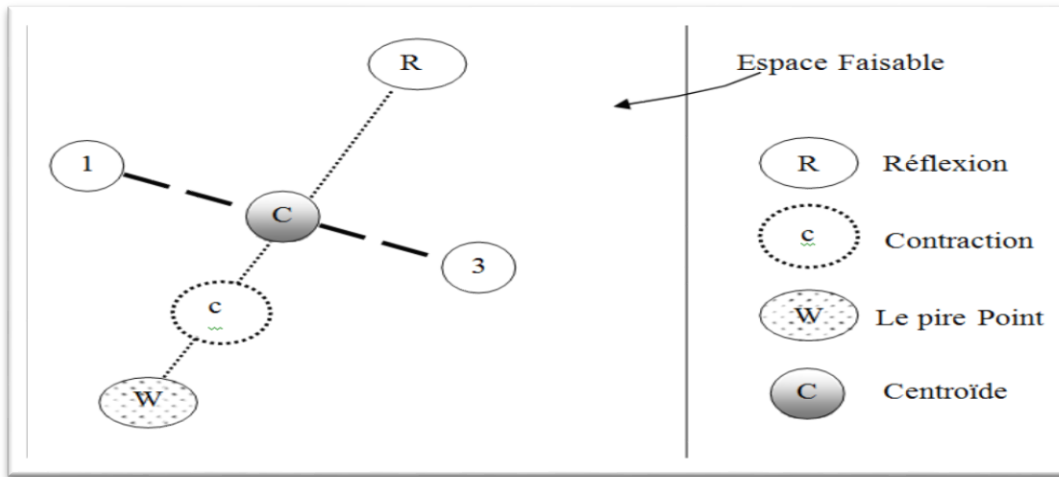


Figure II -10 :Schéma de remplacement de solution en méthode Simplex [32]

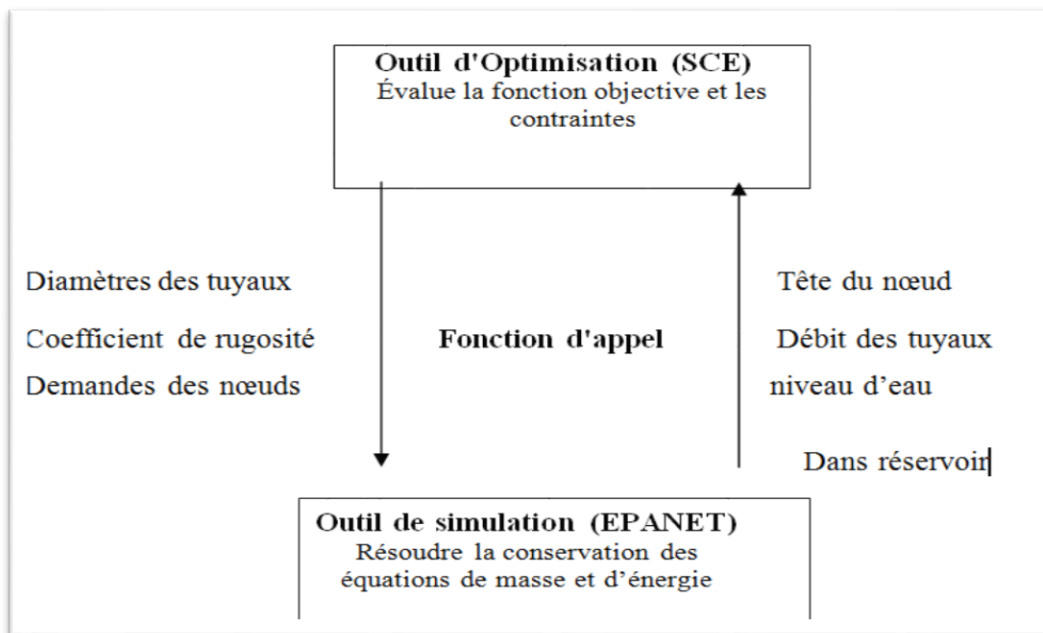


Figure II -11 :Modèles d'optimisation et de simulation [32]

II.2.7.2 langage PYTHON :

Le langage de programmation Python a été créé en 1989 par Guido van Rossum, aux Pays-Bas. Le nom *Python* vient d'un hommage à la série télévisée *Monty Python's Flying Circus* dont G. van Rossum est fan. La première version publique de ce langage a été publiée en 1991.[34]

II.2.7.2.1 La cause de création de python :

À la fin des années 1980, Guido Van Rossum travaillait sur le groupe du système d'exploitation distribué de l'amoeba. Il voulait utiliser un langage interprété comme ABC (ABC a une syntaxe simple et facile à comprendre) qui pourrait accéder aux appels du système amoeba. Ainsi, il a décidé de créer un langage qui était extensible. Cela a conduit à la conception d'un nouveau langage qui a été plus tard nommé Python.[35]

Il est utilisé pour[36]:

- ✓ Développement web (côté serveur),
- ✓ Développement de logiciels,
- ✓ Mathématiques,
- ✓ Système de script.

II.2.7.2.2 Utilisation de Python[36] :

- Python peut être utilisé sur un serveur pour créer des applications Web.
- Python peut être utilisé avec un logiciel pour créer des flux de travail.

- Python peut se connecter aux systèmes de base de données. Il peut également lire et modifier des fichiers.
- Python peut être utilisé pour gérer des données volumineuses et effectuer des calculs complexes.
- Python peut être utilisé pour le prototypage rapide ou pour le développement de logiciels prêts pour la production.
- Utilisé comme langage de script pour automatiser des tâches simples mais fastidieuses,
- On l'utilise également comme langage de développement de prototype lorsqu'on a besoin d'une application fonctionnelle

II.2.7.2.3 Pourquoi Python[36]?

Python est un langage de programmation polyvalent, populaire. C'est une excellente langue maternelle, car elle est concise et facile à lire et écrire par rapport à d'autres langages tels que: C ++, Java, C est un langage puissant. La syntaxe du langage est claire et la longueur du code est relativement courte. C'est amusant de travailler en Python parce qu'il vous permet de réfléchir au problème plutôt que de vous concentrer sur la syntaxe[14]. C'est également un bon langage à avoir dans la pile de programmeurs, car il peut être utilisé pour tout, du développement Web au développement de logiciels et aux applications scientifiques.[38]

- Python fonctionne sur différentes plates-formes (Windows, Mac, Linux, Raspberry Pi, etc.).
- Python a une syntaxe simple semblable à la langue anglaise.
- Python a une syntaxe qui permet aux développeurs d'écrire des programmes avec moins de lignes que certains autres langages de programmation.
- Python fonctionne sur un système interprète, ce qui signifie que le code peut être exécuté dès qu'il est écrit. Cela signifie que le prototypage peut être très rapide.
- Python peut être traité de manière procédurale, orienté objet ou de manière fonctionnelle.
- Extensible et intégrable : Supposons qu'une application nécessite des performances élevées. Vous pouvez facilement combiner des éléments de C / C ++ ou d'autres langages avec du code Python. Cela donnera à votre application des performances élevées ainsi que des fonctionnalités de script que d'autres langages ne fourniront peut-être pas immédiatement.[39]
- Un langage interprété de haut niveau : Contrairement au C / C ++, vous n'avez pas à vous soucier de tâches ardues telles que la gestion de la mémoire, la récupération de place, etc.

De même, lorsque vous exécutez du code Python, il convertit automatiquement votre code dans la langue que votre ordinateur comprend. Vous n'avez pas à vous soucier d'opérations de niveau inférieur.[39]

- Grandes bibliothèques standard pour résoudre des tâches courantes : Python possède un certain nombre de bibliothèques standard, ce qui simplifie grandement la vie des programmeurs, car vous n'avez pas à écrire tout le code vous-même. Par exemple: Besoin de connecter une base de données MySQL sur un serveur Web? Vous pouvez utiliser la bibliothèque MySQLdb avec `import MySQLdb`. Les bibliothèques standard en Python sont bien testées et utilisées par des centaines des personnes. Donc, vous pouvez être sûr que cela ne cassera pas votre demande.[39]
- Orienté objet : Tout en Python est un objet. La programmation orientée objet (POO) vous aide à résoudre un problème complexe de manière intuitive. Avec la programmation orientée objet, vous pouvez diviser ces problèmes complexes en ensembles plus petits en créant des objets.[39]
- Informatique scientifique et numérique : Il existe de nombreuses bibliothèques disponibles en Python pour l'informatique scientifique et numérique. Il existe des bibliothèques telles que: SciPy et NumPy qui sont utilisées dans l'informatique à usage général. Et, il existe des bibliothèques spécifiques telles que: EarthPy pour les sciences de la Terre, AstroPy pour l'astronomie, etc.[37]

En outre, le langage est fortement utilisé dans l'apprentissage automatique, l'exploration de données et l'apprentissage en profondeur.

La version majeure la plus récente de Python est **Python 3.7**, que nous allons utiliser dans notre travail. Cependant, Python 2, bien que n'étant mis à jour avec rien d'autre que des mises à jour de sécurité, reste toujours très populaire.[40]

Dans cette étude, Python sera écrit dans un éditeur de jupyter . Il est possible d'écrire Python dans un environnement de développement intégré, tel que texte, bloc Note, Thonny, Pycharm, Netbeans ou Eclipse, qui sont particulièrement utiles pour gérer des grandes collections des fichiers Python.[40]

*Le carnet de notes Jupyter[41] :

Jupyter Notebook est une application Web à source ouverte qui vous permet de créer et de partager des documents contenant du code en direct, des équations, des visualisations et du texte narratif. Les utilisations incluent: le nettoyage et la transformation des données, la simulation numérique, la modélisation statistique, la visualisation des données, l'apprentissage automatique, etc.

Les mots-clés de Python sont les suivants : and, as, assert, break, class, continue, def, del, elif, else, except, exec, finally, for, from, global, if, import, in, is, lambda, not, or, pass, print, raise, return, try, while, with, yield.

II.2.7.2.4 Syntaxe Python comparée à d'autres langages de programmation :

- Python a été conçu pour la lisibilité et présente certaines similitudes avec l'anglais avec une influence des mathématiques.
- Python utilise de nouvelles lignes pour exécuter une commande, contrairement à d'autres langages de programmation qui utilisent souvent des points-virgules ou des parenthèses.
- Python s'appuie sur l'indentation, en utilisant des espaces, pour définir la portée; tels que la portée des boucles, des fonctions et des classes. D'autres langages de programmation utilisent souvent des accolades.



```
1 ## Imprimir, sumar y contar los numeros que son a la vez multiplos de 2 y
2 ##de 3, que hay entre la unidad y un determinado numero introducido por el
3 ##teclado.
4 n = 1
5 num = input('Llegar hasta: ')
6 h = 0
7 suma = 0
8 while True:
9     if 3*n > num:
10        break
11     if (3*n)%2 == 0:
12        print 3*n,
13        h += 1
14        suma += 3*n
15     n += 1
16 print '\nEntre 1 y %i hay %i numeros multiplos de 3 y de 2' % (num, h)
17 print 'Dichos numeros suman %i' % suma
```

Figure II -12:interface de langage python [41]

II.2.7.2.5 Installer un environnement Python pour Machine Learning avec Anaconda :

L'installation de Python peut-être un vrai challenge. Déjà il faut se décider entre les versions 2.X et 3.X du langage. Et par la suite, choisir les librairies nécessaires (ainsi que les versions compatibles) pour faire du Machine Learning. Sans oublier les subtilités liées aux différents OS (Windows, Linux, Mac...) qui peuvent rendre l'installation encore plus douloureuse.[42]



Figure 0-13 : LOGO D'ANACONDA[42]

II.2.7.2.6 Qu'est ce que Anaconda ?

L'installation d'un environnement Python complet peut-être une vraie galère. Déjà, il faut télécharger Python et l'installer. Par la suite, télécharger un à un les packages dont on a besoin. Parfois, le nombre de ces librairies peut-être grand.

Par ailleurs, il faut s'assurer de la compatibilité entre les versions des différentes packages qu'on a à télécharger.

Anaconda est la plate-forme scientifique des données Python, c'est une distribution Python. A son installation, Anaconda installera Python ainsi qu'une multitude de packages (voir liste des packages anaconda). Cela nous évite de nous ruer dans les problèmes d'incompatibilités entre les différents packages.

Finalement, Anaconda propose un outil de gestion de packages appelé **Conda**. Ce dernier permettra de mettre à jour et installer facilement les librairies dont on aura besoin pour nos développements. [42]

❖ Téléchargement d'Anaconda

Note : Les instructions qui suivent ont été testées sur Windows 10. Le même processus d'installation pourra s'appliquer pour les autres OS. [42]

- Pour télécharger Anaconda, rendez -vous à la [page de téléchargement](#).
- Prenez la version du binaire qu'il vous faut :
 - Choisissez le système d'exploitation cible (Windows, Mac, etc...)
 - Sélectionnez la version 3.X (à l'heure de l'écriture de ces lignes, c'est la version 3.7 qui est proposée)
 - Si possible, privilégiez la version 64 bits.



Figure II -14: les versions python installer [42]

Le fichier binaire à télécharger fait environ 662 Mo.

❖ Installer Anaconda

L'installation d'anaconda se fait assez naturellement, il suffit de double cliquer sur le fichier exécutable pour lancer l'installation.

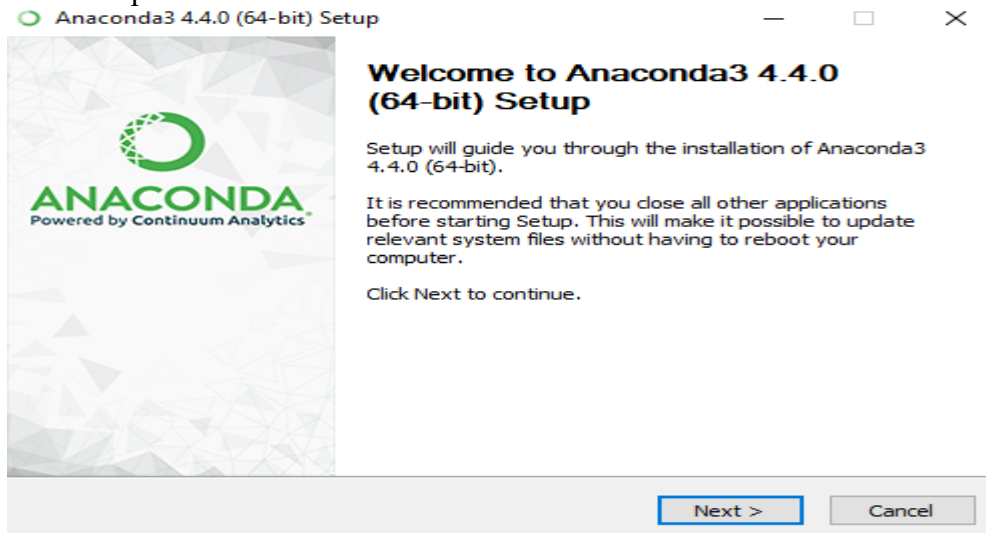


Figure 0-15: instalation de ANACONDA [42]

Lancement et mise à jour d'Anaconda

A l'issue de l'installation, vous aurez **Anaconda Navigator**. Ce dernier est une interface graphique permettant de lancer *Jupyter* (pour écrire du python au format *notebook*), ou encore *Spyder* (IDE pour éditer du code python).[42]

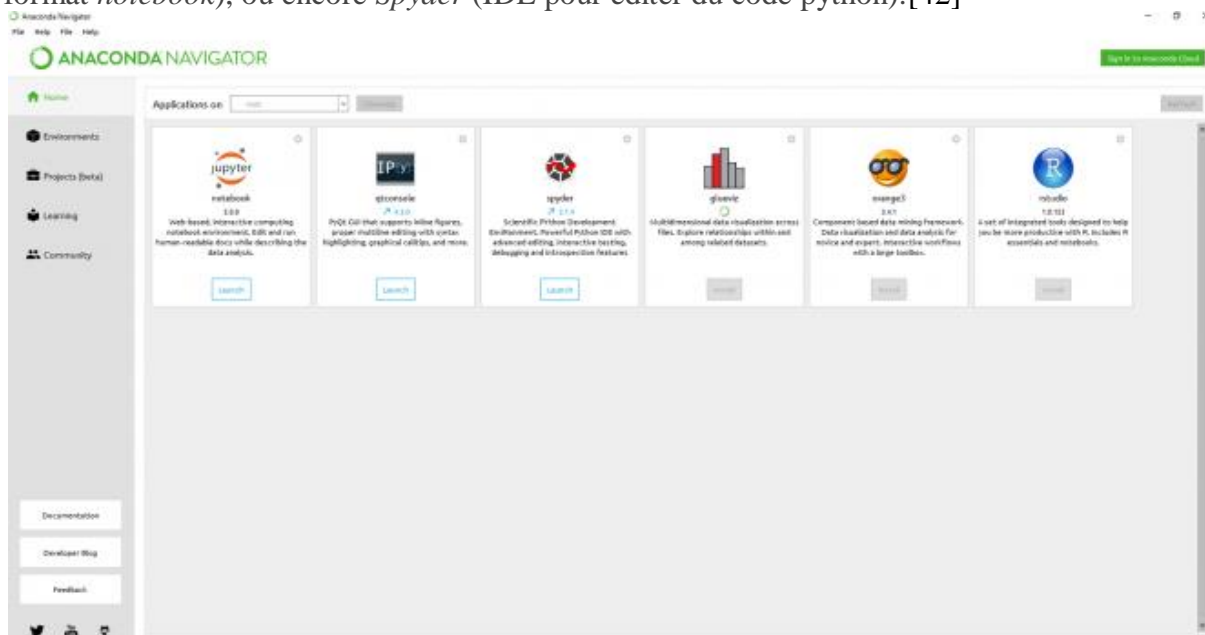


Figure 0-16: Les Déférente IDE dans ANACONDA [42]

Depuis cette interface graphique, il est tout à fait possible de gérer les packages installés.

II.2.7.2.7 Visuelle studio :

Visual Studio est un **environnement de développement intégré (IDE)** développé par Microsoft pour développer une interface utilisateur graphique (GUI), une console, des applications Web, des applications mobiles, un cloud et des services Web, etc. Avec cet IDE, vous pouvez créer du code géré ainsi que du code natif. Il utilise les différentes plates-formes de logiciels de développement logiciel Microsoft, telles que Windows Store, Microsoft Silverlight, Windows API, etc. Il ne s'agit pas d'un IDE spécifique à la langue, car vous pouvez l'utiliser pour écrire du code en C #, C ++, VB (Visual Basic), Python, JavaScript et beaucoup d'autres langues. Il fournit un support pour 36 langages de programmation différents. Il est disponible pour Windows ainsi que pour macOS.[43]

a) Mise en route de Visual Studio 2017 :

Tout d'abord, nous devons télécharger et installer Visual Studio 2017 . N'oublions pas de sélectionner le workload principal .NET lors de l'installation de VS 2017.

Nous pouvons voir un certain nombre de fenêtres d'outil lorsque nous ouvrons Visual Studio et commencez à écrire notre premier programme comme suit:

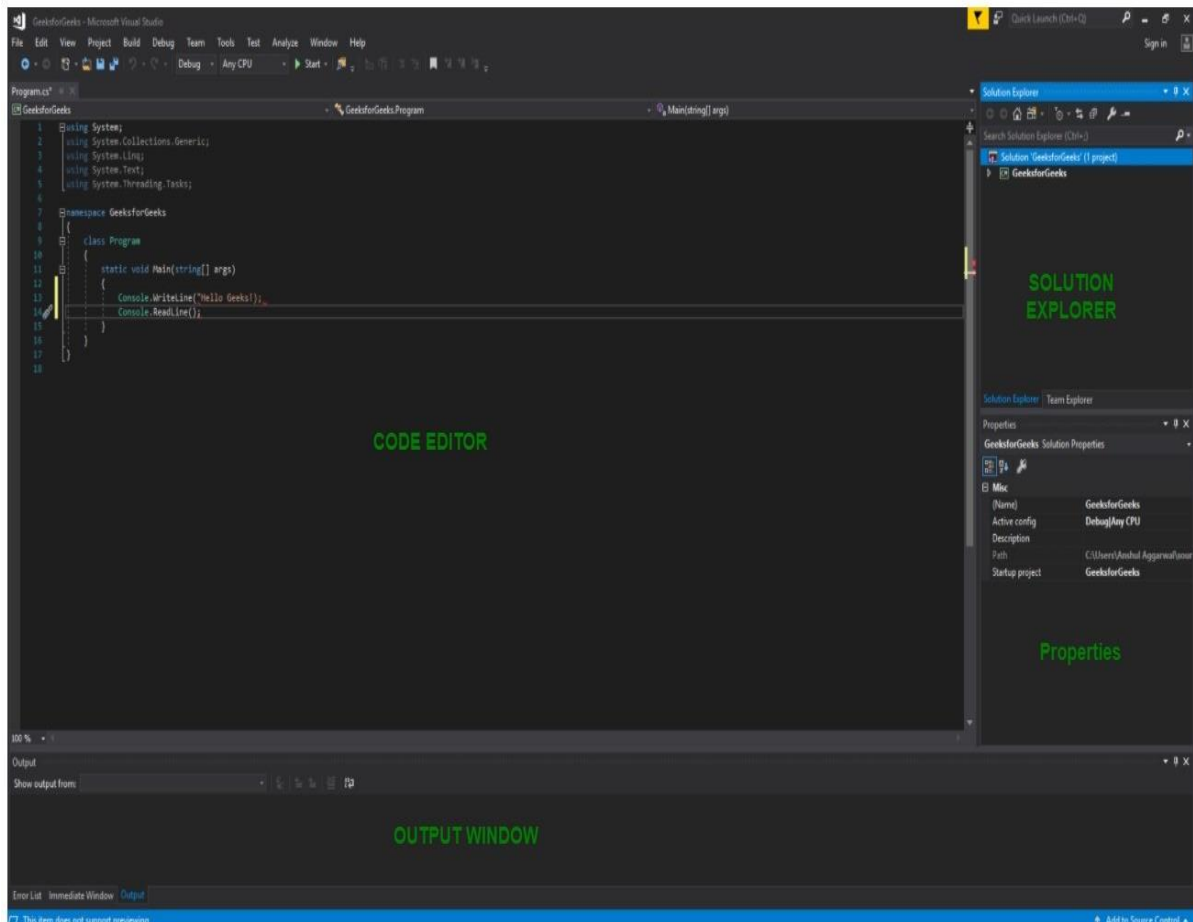
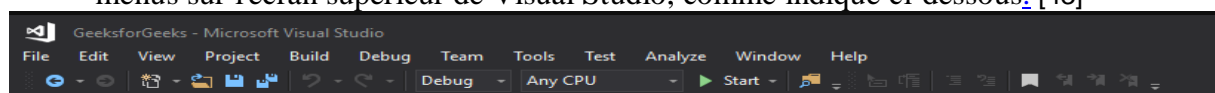


Figure 0-17 : interface de visual studio [43]

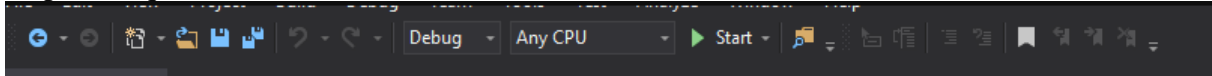
1. **Editeur de code:** où l'utilisateur écrira du code.
2. **Fenêtre de sortie:** Visual Studio affiche les sorties, les avertissements du compilateur, les messages d'erreur et les informations de débogage.
3. **Explorateur de solutions:** Affiche les fichiers sur lesquels l'utilisateur travaille actuellement.
4. **Propriétés:** Il donnera des informations supplémentaires et un contexte sur les parties sélectionnées du projet en cours.
 - ❖ Un utilisateur peut également ajouter des fenêtres selon les besoins en les choisissant dans le menu Affichage (**view**). Dans Visual Studio, les fenêtres d'outil sont personnalisables puisqu'un utilisateur peut ajouter plus de fenêtres, supprimer celles qui sont déjà ouvertes ou déplacer les fenêtres à sa convenance. [43]
 - ❖ Différents menus dans Visual Studio: un utilisateur peut trouver un grand nombre de menus sur l'écran supérieur de Visual Studio, comme indiqué ci-dessous. [43]



- 1) Les commandes créer, Ouvrir et Enregistrer des projets sont contenues dans le menu **File**.
- 2) Les commandes de recherche, de modification et de refactorisations sont contenues dans le menu **Edit**.
- 3) Le menu **View** permet d'ouvrir les fenêtres d'outils supplémentaires dans Visual Studio.

- 4) Le menu **Project** permet d'ajouter des fichiers et des dépendances au projet.
- 5) Pour modifier les paramètres, ajoutez des fonctionnalités à Visual Studio via des extensions et accédez à divers outils Visual Studio pouvant être utilisés à l'aide du menu **Outils**.

Le menu ci-dessous est appelé la **barre d'outils** qui fournit un accès rapide aux commandes les plus fréquemment utilisées. [43]



CONCLUSIONS

Dans ce chapitre nous avons présenté les principes application de l'algorithme et EPANET et leur domaine d'utilisation.

Un schéma basé sur le couplage SCE et EPANET est créé. Des suggestions sur la manière de rendre le système plus robuste et plus efficace sont données. Un schéma robuste et efficace est essentiel pour la conception des réseaux de distribution d'eau et la réhabilitation des réseaux existants. Il a été constaté que le système proposé est capable de résoudre à la fois les petits et les grands réseaux. Il recherche avec des diamètres initialement aléatoires assignés à toutes les canalisations du réseau et produit finalement un coût minimum avec des diamètres optimaux de canalisations satisfaisant toutes les contraintes hydrauliques prescrites. Le détail de leur application sera abordé dans le chapitre suivant.

Chapitre -I I I -

APPLICATIONS DU SCHÉMA PROPOSÉ POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

III INTRODUCTION :

Les problèmes les plus courants dans les systèmes de distribution d'eau sont la conception des nouveaux réseaux de distribution d'eau ou les extensions ou rénovation d'un réseau déjà existant. L'objectif principal est de satisfaire la demande en eau des consommateurs à des hauteurs de pression spécifique, même en cas de rupture de conduite. Il est en effet difficile d'assurer la fiabilité économique d'un système de réseau. Si le réseau est dimensionné avec un ensemble des tuyaux sous-dimensionnés, la pression minimale requise peut ne pas être satisfaisante. Par contre, si la conduite du réseau est surdimensionnée, la conception peut ne pas être économiquement réalisable. La recherche s'est concentrée sur le développement d'un schéma adapté permettant de déterminer le coût le plus bas pour le réseau d'alimentation en eau qui répond à toutes les contraintes hydrauliques. Ce chapitre décrit (1) la conception et la formulation d'un problème d'optimisation du réseau des canalisations suivi de ses applications (SCE-EPANET) sur la conception d'un nouveau réseau ; et la réhabilitation d'un réseau existant.

III.1 CONCEPTION ET FORMULATION :

L'objectif de la conception d'un réseau de distribution d'eau est de trouver le diamètre optimal de chaque conduite de réseau pour un tracé donné, demande de base et une politique d'opération. Le modèle sélectionne les tailles optimales du réseau final en tenant compte de toutes les contraintes implicites et explicites telles que la conservation de la masse, la conservation de l'énergie, les contraintes de la charge et de conception. Les contraintes implicites impliquent la conservation de la masse et la conservation de l'énergie alors que les contraintes explicites incluent les contraintes hydrauliques et les contraintes de diamètre. Les contraintes hydrauliques concernent la pression au nœud qui doit être supérieure à une valeur minimale spécifique. Si la contrainte hydraulique n'est pas satisfaite, une fonction de pénalité est introduite dans la formulation. Cependant, les contraintes de diamètre limitent les algorithmes évolutifs pour sélectionner la solution d'essai à l'intérieur d'une limite prédéfinie. Un solveur de réseau hydraulique gère les contraintes implicites et évalue la performance hydraulique de chaque solution d'essai qui est un membre d'une population des points. Les informations hydrauliques obtenues à partir du solveur de réseau sont utilisées pour le calcul de la fiabilité de la conception. La fiabilité d'une solution d'essai représentant la conception d'un réseau de canalisation est basée sur les performances hydrauliques obtenues à partir du solveur de réseau. Il se compose de deux parties. L'un est le coût du réseau et l'autre est le coût de la pénalité, comme il est expliqué dans les sections suivantes.[1]

III.1.1 Coût du réseau :

Le coût du réseau est pris comme la somme des coûts des canalisations exprimés en termes de coût par unité de longueur. Il est possible d'entrer le coût individuel pour chaque conduite ou d'utiliser l'équation de coût pour calculer le coût du réseau. Toutefois, les variables de conception, à savoir les dimensions des tuyaux, doivent être choisies parmi un ensemble des diamètres distincts disponibles dans le commerce. Certains modèles considèrent les diamètres comme une variable continue. Mais ce modèle ne garantit pas la qualité de la solution finale en raison de la conversion aux dimensions les plus proches du marché des tuyaux. Dans cette étude, les valeurs réelles des variables générées par SCE sont immédiatement converties en diamètres commerciaux pour calculer le coût du réseau. La procédure est la suivante :

- ❖ Les diamètres des canalisations commerciales sont triés par ordre croissant.
- ❖ Les coûts pour toutes les tailles des tuyaux disponibles sont indiqués dans le fichier d'entrée.
- ❖ Lisez la taille et le coût du marché dans le tableau.
- ❖ Soit $D(j)$ et $D(k)$ représentent respectivement les variables générées par la SCE et la taille du marché des tuyaux.
si $D(j) < D(k+1)$ et $D(j) \geq D(k)$ alors $D(k)$
si $D(j) \leq (D(k+1) + D(k))/2$ alors $D(j) = D(k)$

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA REHABILITATION DES RESEAUX DE DISTRIBUTION D'EAU

sinon

$$D(j) = D(k+1)$$

fin si

fin si

Le coût du réseau dépend du prix des tuyaux du marché et il est calculé à l'aide de l'équation suivante :

$$C_N = \sum_{k=1}^{NP} ck(Dk).Lk \dots\dots\dots(III.1)$$

où $ck(Dk)$ = coût par unité de longueur du $k^{i\text{ème}}$ tube de diamètre Dk ;

Lk = longueur du $k^{i\text{ème}}$ tuyau ; et NP = nombre total des tuyaux dans le système.

- Le coût total comprend le coût de la canalisation, le coût de la pénalité de vitesse et le coût de la pénalité de pression. La formule du coût total utilisée dans l'algorithme du programme est présentée ci-dessous ;

III.1.2 Fonctions de pénalité :

Le terme de pénalité est déclenché dans la fonction objectif lorsqu'une solution ne répond pas à la ou aux contraintes, qui sont, par exemple, les exigences de pression minimale. Si une conception de réseau ne satisfait pas les exigences minimales de pression, un coût supplémentaire exprimé sous forme de coût de pénalité s'ajoute au coût réel du réseau. Différentes méthodes de pénalité ont été proposées. Le coût de la pénalité pour tout réseau de conception irréalisable est calculé sur la base du degré de violation de la pression. La violation de pression (ΔH_j), à n'importe quel noeud (j), est la différence entre la pression minimale ($H_{\min,j}$) et la pression simulée (H_j).

Le déficit de pression maximale est souvent utilisé dans de diverses études pour calculer le coût de la pénalité (Simpson et al., 1994 ; Savic et Walters, 1997). Ainsi, l'équation peut être exprimée comme suit,

$$C_p \text{ (coût de pénalité)} = p_c \times \max_j \Delta H_j = p_c \times \max_j [\max (H_{\min,j} - H_j, 0)] \dots\dots\dots(III.2)$$

où p_c est le facteur de pénalité.

En 1998, deux équations de pénalité différentes ont été proposées par Abebe et Solomatine pour deux conditions différentes. Si la pression est inférieure à zéro, un coût de pénalité élevé est ajouté au coût réel du réseau comme suit :

$$C_{p1} = 2 \times p_{c1} \times C_{\max} - 2 \times C_N \dots\dots\dots(III.3)$$

Si la pression est toutefois inférieure à la limite minimale et supérieure à zéro, l'équation considérée est :

$$C_{p2} = p_{c2} \times C_{\max} \times \max_{j=1 \text{ à } N_j} (H_{\min,j} - H_j) \dots\dots\dots(III.4)$$

où p_{c1} et p_{c2} sont les coefficients de coût de pénalité;

C_{\max} est le coût maximal possible calculé à partir du plus grand tuyau commercial disponible;

et $(H_{\min,j} - H_j)$ est le déficit de pression au noeud j .

Les équations (III.2) et (III.4) ne tiennent compte que du déficit de pression maximale et négligent des déficits de pressions aux autres noeuds. Cela signifie que dans le cas de violation égale de la pression maximale, le même coût de pénalité est facturé pour les deux solutions irréalisables. Pour incorporer les violations de pression à d'autres noeuds, la somme des déficits de pression de tous les noeuds est introduite dans le coût de la pénalité. Ainsi, la fonction de coût de pénalité devient :

$$C_p = p_c \times \sum_{j=1}^{N_j} \Delta H = p_c \times \sum_{j=1}^{N_j} (H_{\min,j} - H_j) \dots\dots\dots(III.5)$$

où N_j = nombre total des jonctions (les noeuds) dans le système où il y a violation de la pression.

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

Le coefficient de coût de pénalité (p_c) dans toutes les fonctions de pénalité décrites ci-dessus dépend de la structure du problème et doit être choisi avec soin. Le choix d'un facteur de pénalité approprié a été un problème parce que le facteur de pénalité est trop petit, conduit la recherche dans la région infaisable, alors que trop grand facteur de pénalité impose la recherche vers la région d'Optima local et empêche strictement l'utilisation de la solution presque optimale. Dans cette étude, le facteur de pénalité est considéré comme **une variable**. L'algorithme optimise sa valeur à l'intérieur d'une limite de plage spécifique. Étant donné que les différents points de la population ont de différents degrés de faisabilité, le point individuel devrait avoir son propre coefficient de pénalité. Cette approche permet d'adapter automatiquement le facteur de pénalité et d'éliminer la difficulté de la sélection des essais.

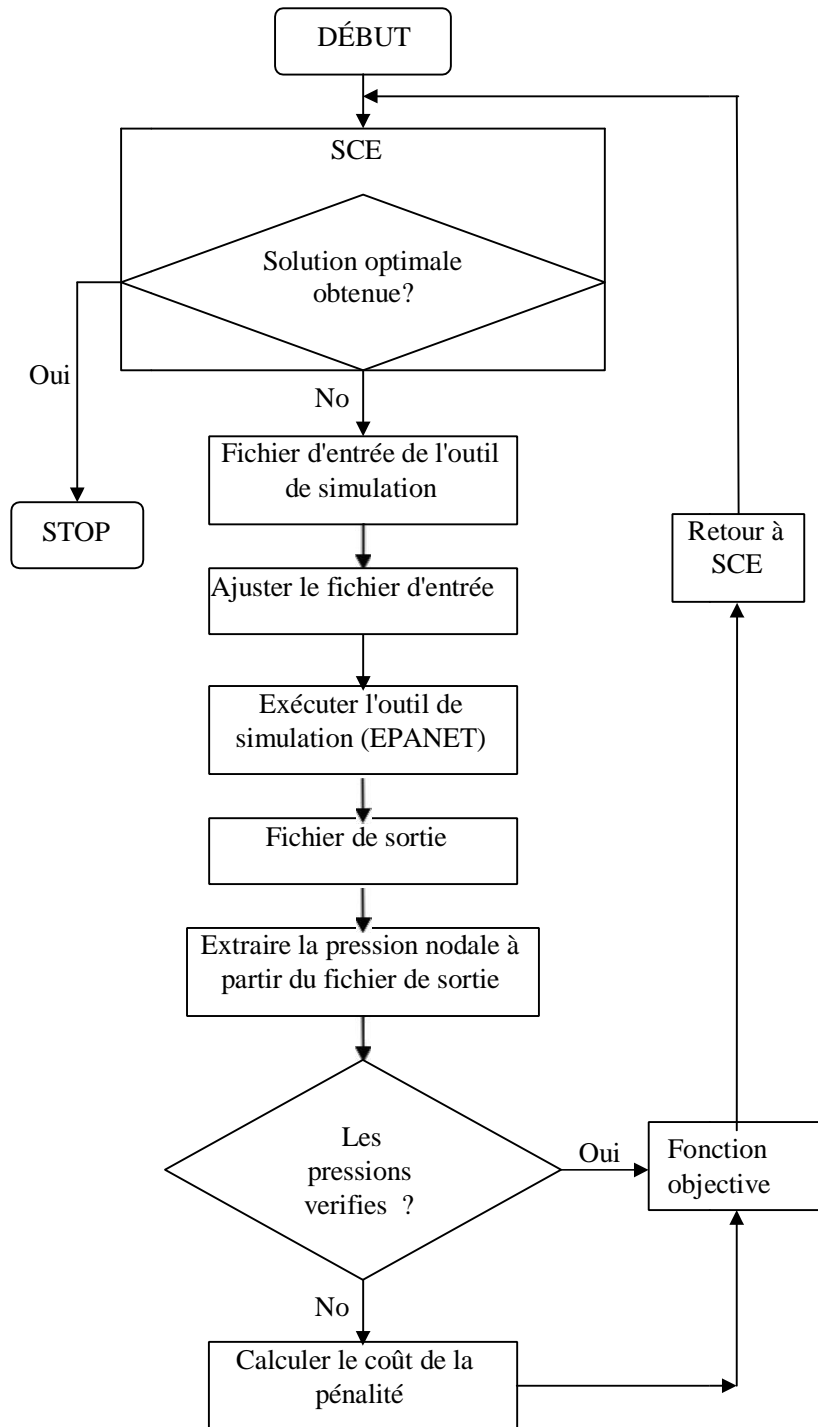


Figure III-1: Organigramme du problème de conception

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

III.1.3 Contraintes :

Dans la conception et l'optimisation du réseau de distribution d'eau, les contraintes couramment considérées sont résumées ci-dessous :

1. Contraintes de diamètre : les diamètres des tuyaux doivent être choisis parmi les dimensions commerciales disponibles. Dans cette étude, les tailles générées par la SCE sont immédiatement converties aux tailles de marché disponibles les plus proches pour répondre à ces contraintes.
2. Contraintes hydrodynamiques : il s'agit notamment de la conservation de la masse et de l'énergie. Normalement, le modèle de simulation de réseau EPANET contrôle ces contraintes.
3. Contraintes de pression : La pression hydraulique à chaque nœud du réseau doit être supérieure ou égale à une valeur prédéfinie. La violation de la pression due à une solution infaisable est expliquée par la pénalisation du surcoût du réseau selon l'équation (III.5).

La formulation peut être mathématiquement exprimée comme suit :

Minimiser le coût $C = C_N + C_P$ (III.6)

Soumis à :

$$\sum Q_j = 0 \quad \text{Conservation de l'équation de masse} \quad (3.7a)$$

$$\sum H_L = \sum P_E \quad \text{Équation de conservation de l'énergie} \quad (3.7b)$$

$$H_j \geq H_{\min, j} \quad \text{Contraintes de pression nodale} \quad (3.7c)$$

$$D_k \in \{D_1, D_2, D_3, D_4, \dots, D_n\} \quad (3.7d)$$

$$D_{\min} \leq D_k \leq D_{\max} \quad (3.7e)$$

$$p_{cl} \leq p_c \leq p_{cu} \quad (3.7f)$$

où

Q_j = débit entrant ou sortant du nœud j ;

H_L = perte de charge dans la conduite ;

P_E = énergie de pompage appliquée ;

D_k = variables de décision (dimensions de la conduite) ;

D_{\min} = taille minimale du marché ;

D_{\max} = taille maximale du marché ;

$\{D_1, D_2, D_3, D_4, D_4, \dots, D_n\}$ = ensemble des tailles commerciales disponibles ;

p_{cl} = la borne inférieure du facteur de pénalité et p_{cu} = la borne supérieure du facteur de pénalité.

III.1.4 MÉCANISME DE FONCTIONNEMENT DU MODÈLE SCE-EPANET

Une brève description des étapes du SCE-EPANET pour l'optimisation du réseau de canalisations est donnée ci-dessous et illustrée à la figure III.1:

1. Génération d'une population des points par SCE. Chaque point représente une combinaison des diamètres des conduites de canalisation du réseau.
2. Calcul le coût du réseau pour chacune des solutions après la conversion des tailles des conduites générées au hasard aux tailles du marché.
3. Lire les données du réseau à partir du fichier d'entrée de l'outil de simulation.
4. Ajuster le diamètre du tuyau dans le fichier d'entrée.
5. Effectuer l'analyse hydraulique de chaque réseau avec le solveur de réseau EPANET.
6. Lire la pression nodale à partir du fichier de sortie d'EPANET. Vérifier la pression à certains nœuds pour répondre à certaines pressions nodales.
7. Calculer le coût de la pénalité si la hauteur de pression à n'importe quel nœud est inférieure au minimum requis.
8. Calculer la fonction objective qui est la somme du coût du réseau et du coût de la pénalité trouvés aux étapes 2 et 7 respectivement.
9. Le coût total indiqué à l'étape 8 est utilisé comme valeur de critère pour chacun des réseaux d'essai.

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

10. Les critères d'arrêt sont vérifiés, si l'un des critères de convergence est satisfaisant, le processus est terminé.

III.1.5 CRITÈRES D'ARRÊT UTILISÉS DANS LA CONCEPTION DU RÉSEAU :

Dans la conception du réseau des canalisations, deux critères d'arrêt sont vérifiés à chaque génération. Si l'un d'entre eux est satisfaisant, l'algorithme de recherche s'arrête. Les critères sont les suivants :

- 1) Le changement relatif de la fonction objective dans les **k** dernières boucles de shuffling n'a pas changé plus qu'un pourcentage prédéterminé (1%).
- 2) Le nombre maximum d'évaluations de fonctions spécifié par l'utilisateur est atteint.

III.1.6 Fonction objective et les contraintes :

Minimiser le coût total C_T

$$\begin{aligned} \text{Soumis à : } & (TH)_i \geq (TH_{\min})_i & i=1,2,3,\dots & N & \dots & (1) \\ & V_{\min} \leq V_j \leq V_{\max} & j=1,2,3,\dots & M & \dots & (2) \\ & NRP \geq NRP_{\min} & & & \dots & (3) \\ & \sum Q_k = 0 & k=1,2,3,\dots & P & \dots & (4) \\ & \sum h_x = 0 & x=1,2,3,\dots & R & \dots & (5) \\ & D_j \geq 0 & & & \dots & (6) \end{aligned}$$

Où N= Nombre s nœuds dans le réseau ;

M= Nombre des tronçons dans le réseau ;

P= Nombre des tronçons reliés au nœud ;

R= Nombre des tronçons dans la maille ;

C_T = coût total du réseau ;

TH = la pression au nœud en m ;

THmin = HGL la pression minimale requis au nœud ;

Vmin = Vitesse d'écoulement minimale dans le tronçon en m/s (0,3 m/s) ;

Vmax = Vitesse d'écoulement maximale dans le tronçon en m/s (2.5 m/s) ;

V= Vitesse réelle du débit dans le tronçon en m/s ;

NRP= Paramètre de fiabilité du réseau ;

NRPmin = Paramètre de Fiabilité minimale du réseau;

III.1.7 Paramètres d'optimisation SCE :

Il existe une série des paramètres d'optimisation SCE. Ils le sont :

n = nombre des paramètres;

p = nombre des complexes (p 1);

pmin = nombre minimal des complexes requis, si le nombre des complexes peut être diminué au fur et à mesure que l'optimisation avance ($1 \leq p_{\min} \leq p$)

m = nombre des points dans chaque complexe de la population initiale ($m \geq 2$)

q = nombre des points dans chaque sous-complexe ($1 \leq q \leq m$)

α = nombre des progénitures générées par chaque sous-complexe

β = nombre d'étapes d'évolution autorisées pour chaque complexe avant le mélange complexe,

Taille de la population = p x m.

III.1.8 ÉTUDES DE CAS :

Pour vérifier la fiabilité de notre programme, on va utiliser le réseau dimensionné auparavant dans mon mémoire de fin d'étude session 2018.

III.1.8.1 Présentation du Réseau d'AEP de la zone d'étude :

Le réseau de la zone basse de la commune de SIDI LAKHDER est un réseau maillé. Le réseau de distribution sera constitué d'un assemblage de tuyaux en PEHD qui résiste à une pression nominale de 10 bars (PN=10bar), les diamètres utilisés varient entre DN 90 mm et DN 400 mm avec des pressions varient entre 24,11m et 44,93 m.

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

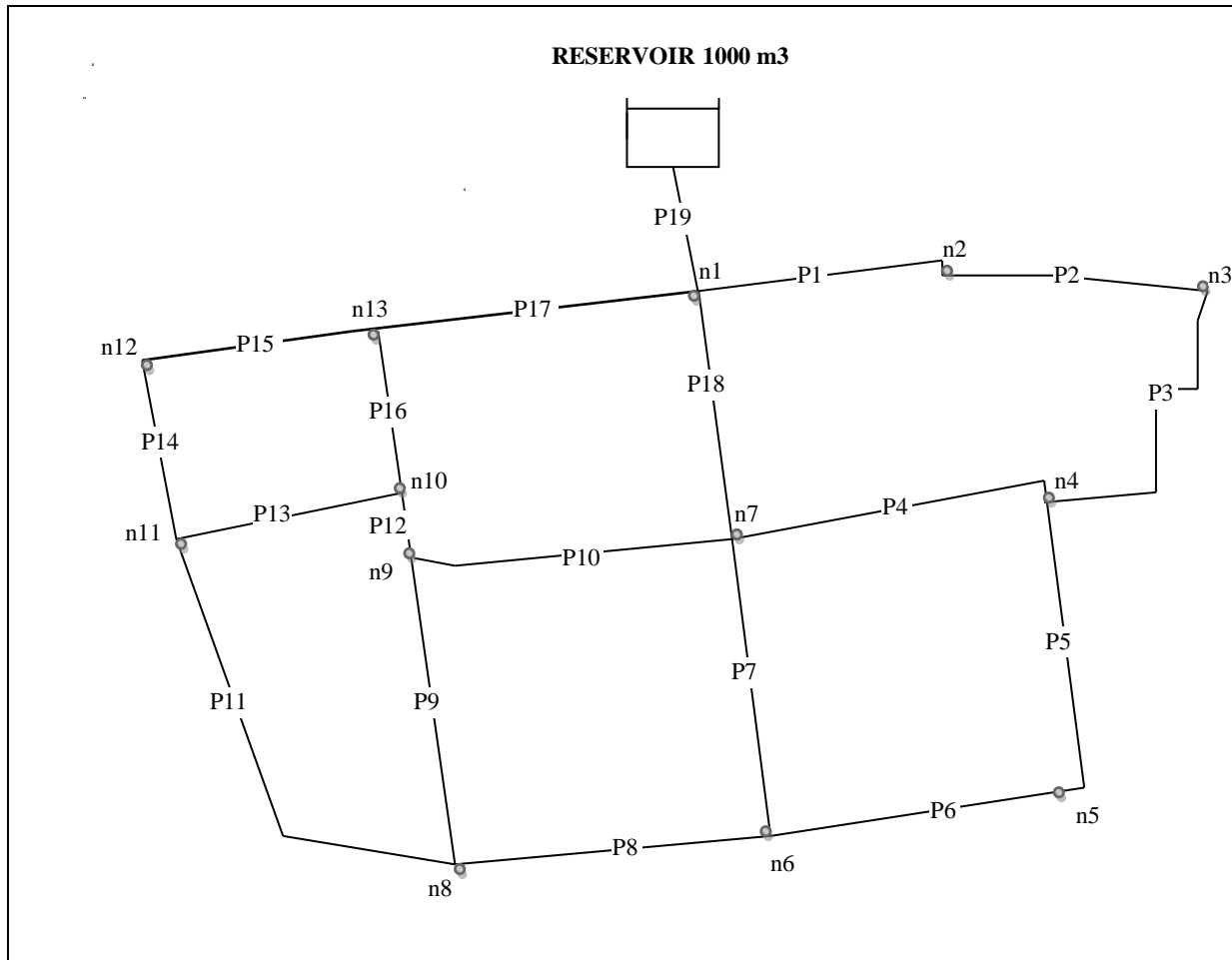


Figure III-2: Réseau d'AEP de notre zone d'étude

Le détail de dimensionnement des différents tronçons est mentionné dans le tableau III.1. et le coût dans le tableau III.2

Tableau III-1: Les Différents Tronçons Avec Leurs Caractéristiques Hydrauliques Et Géométriques

tronçons	Longueur (m)	Diamètre intérieure (mm)	Rugosité (mm)	Débit (l/s)	Vitesse (m/s)
P19	576,35	352,6	0,015	190,18	1,95
P1	341,6	176,2	0,015	30,06	1,23
P18	323	277,6	0,015	90,18	1,49
P17	425,9	220,4	0,015	54,39	1,43
P2	379	176,2	0,015	19,78	0,81
P3	455	141	0,015	7,88	0,5
P5	451,3	141	0,015	15,7	1,01
P4	445,6	176,2	0,015	27,11	1,11
P6	360,7	110,2	0,015	4,11	0,43
P7	419,82	141	0,015	24,69	1,58
P8	420,7	141	0,015	11,66	0,75
P10	427	141	0,015	15,33	0,98
P9	456,6	96,8	0,015	7,84	1,07
P11	680,3	79,2	0,015	2,74	0,56
P12	89,78	96,8	0,015	6,4	0,87
P13	298,09	96,8	0,015	7,13	0,97

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA REHABILITATION DES RESEAUX DE DISTRIBUTION D'EAU

P16	10_13	197,1	176,2	0,015	21,89	0,9
P14	11_12	190	141	0,015	12,27	0,79
P15	12_13	302,4	141	0,015	19,3	1,24

Tableau III-2: Coûts des différents diamètres (avant optimisation)

Les diamètres (mm)	UNITE	QUATITE	PRIX UNITAIRE (DA)	UNITE (DA)
90	ML	680,3	420,14	285821,24
110	ML	844,47	632,39	534034,38
125	ML	360,7	772,36	278590,25
160	ML	2666,22	1336,43	3563216,39
200	ML	1363,33	1931,48	2633244,63
250	ML	425,9	3036,15	1293096,29
315	ML	323	4789,47	1546998,81
400	ML	442,54	7837,53	3468420,53
250	ML	1379,35	3036,15	4187913,50
Coût Total				17791336,02

III.1.8.2 Application de notre programme et interprétation :

The screenshot shows the GANEO software interface. The main window is titled "GENETIC ALGORITHM NETWORK OPTIMISATION (GANEO)". It features a menu bar (File, Option, Info, Help) and a toolbar with icons for file operations. The interface is divided into several panels:

- EPANET Network:** Contains buttons for "Load file", "View file", and "View network". Below these is the filename "epanet drici.inp" and a table of pipe data with columns: Pipe ID, Start node, End node, Length, Diameter, Roughness, and Fixed.
- Boundaries:** Includes input fields for "Minimum required residual pressure" (set to 24 m) and "Maximum velocity" (set to 2.5 m/s).
- GA parameters:** Includes settings for "Cross-over method" (Single point), "Mutation" (Random), "Mutation percentage" (2.00%), "Termination" (Maximum generatic), "Elapsed time" (1000 seconds), "No of iterations" (100), "Number of favourable solutions" (20), and "Population size" (10).
- Network Description:** Features radio buttons for "New network", "Improvement (Additional pipes)", and "Improvement (Replacing pipes)".
- Pipe data:** Includes buttons for "Create file" and "Load file", and a table of pipe specifications with columns: Pipe name, Class, Nominal diameter, Inside diameter, Roughness, and Cost.
- User comments:** A text area for entering user comments.
- Run GA optimization:** A button at the bottom right to execute the optimization, accompanied by a help icon.

Figure III-3: interface de programme d'optimisation (GANEO)

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA REHABILITATION DES RESEAUX DE DISTRIBUTION D'EAU

Notre programme d'optimisation nécessite les données de réseau hydraulique préparé par Epanet, qui sont ajuster dans le format d'exportation d'EPANET appelé fichier INP. Tous les données du réseau et les informations sont conservé dans la bibliothèque DLL de EPANET sera trouvé dans sa code source qui on a déjà téléchargé. Et aussi on télécharge autre fichier les prix des conduites en format pip.

On lance le programme pour exécution on 100 itérations. Il donne plus de 1000 solutions .le programme traite automatiquement les solutions optimales. Les résultats optimaux obtenus par notre programme d'optimisation sont représentés dans le tableau III.3 et le tableau III.4.

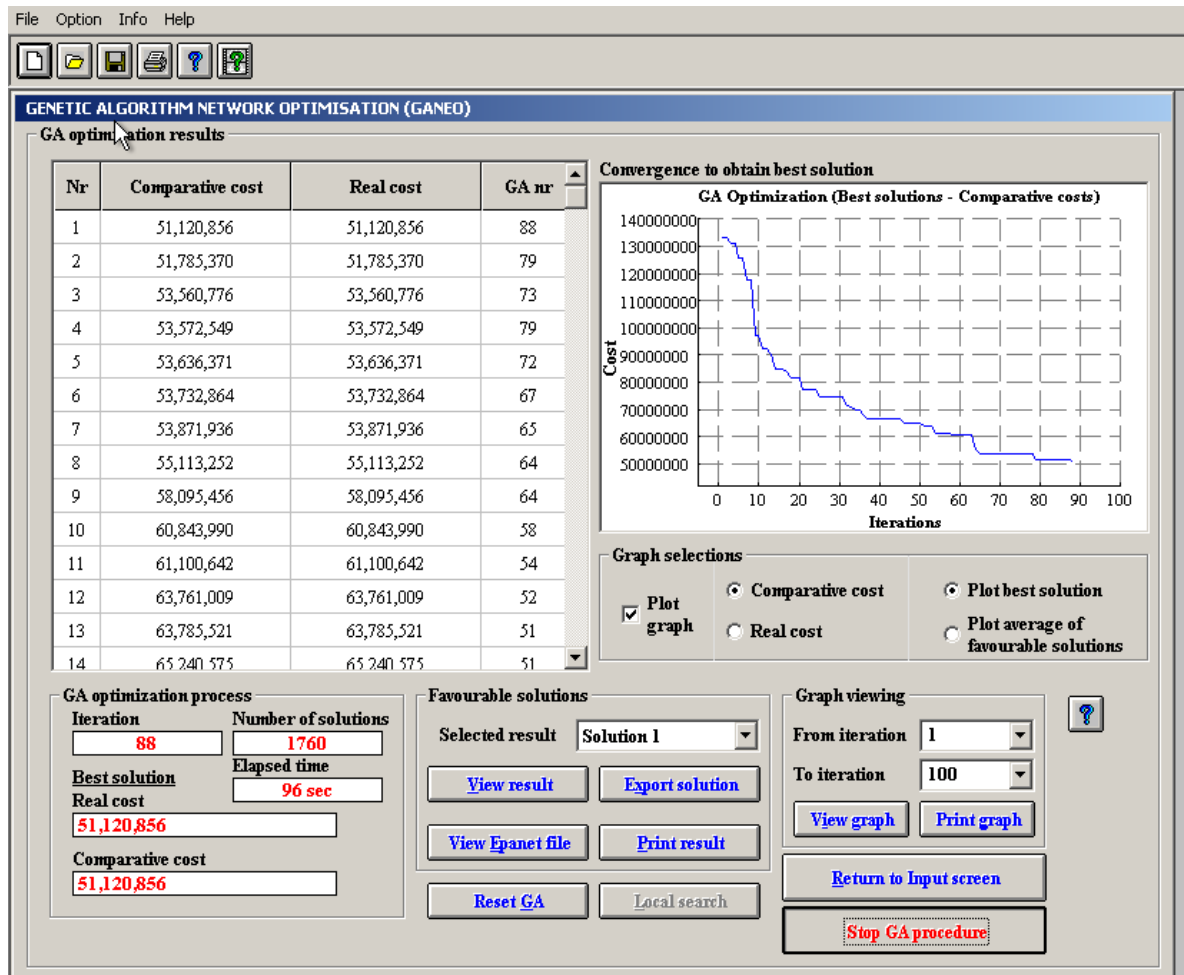


Figure III-4: Exécution du programme

Vous avez remarqué que durant exécution le cout de réseau diminue à chaque itération jusqu'à atteindre un certain limite se que n'appel une valeur optimale. La simulation prendre quelque seconde. On prend la première solution.

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA RÉHABILITATION DES RÉSEAUX DE DISTRIBUTION D'EAU

Tableau III-3: le dimension des différents diamètres de réseau (après optimisation)

	tronçons	Longueur (m)	Diamètre intérieure (mm)	La vitesse (m/s)
P19	R_1	576,35	440.6	1.03
P1	1_2	341,6	352,6	0.95
P18	1_7	323	277.6	0.69
P17	1_13	425,9	176.2	0.91
P2	2_3	379	352,6	1.03
P3	3_4	455	110,2	1.2
P5	4_5	451,3	79,2	1.65
P4	4_7	445,6	220,4	0.87
P6	5_6	360,7	141	0.93
P7	6_7	419,82	277.6	0.62
P8	6_8	420,7	141	0.77
P10	7_9	427	27,2	0.9
P9	8_9	456,6	53,6	0.5
P11	8_11	680,3	79,2	1.43
P12	9_10	89,78	220,4	0.68
P13	10_11	298,09	96.8	1.12
P16	10_13	197,1	141	0.82
P14	11_12	190	96.8	0.86
P15	12_13	302,4	79,2	0.74

Lorsqu'on examine le tableau III-4, les diamètres trouvés par notre programme sont les mêmes que le meilleur résultat. De plus, le temps de calcul pour obtenir ce résultat est très court ce qui environ 1 minute.

Tableau III-4: Pressions à divers nœuds (après optimisation)

Les nœuds	La pression (m)
n01	39.31
n02	30.52
n03	24.31
n04	28.34
n05	25.82
n06	28.96
n07	37.37
n08	27.44
n09	30.76
n10	30.41
n11	32.80
n12	29.41
n13	35.66

CHAPITRE III : APPLICATIONS DU SCHEMA PROPOSE POUR LA CONCEPTION ET LA REHABILITATION DES RESEAUX DE DISTRIBUTION D'EAU

On peut voir qu'il n'y a pas de pression inférieure à 20 mètres, ce qui est le critère le plus important du notre programme.

Tableau III-5: Coûts des différents diamètres (après optimisation)

Le diamètre (mm)	QUANTITE (ML)	PRIX UNITAIRE (DA)	UNITE (DA)
440,6	576,35	11878,44	5647724,12
352,6	720,6	7837,53	3557714,11
277,6	742,82	4789,47	1625493,99
220,4	535,38	3036,15	822617,332
176,2	425,9	1931,48	1307696,76
141	978,5	1336,43	351423,8
110,2	455	772,36	308663,235
96,8	488,09	632,39	602480,76
79,2	1434	420,14	112022,244
53,6	456,6	245,34	1765,552
27,2	27,2	64,91	1765,552
Coût Total			14339367,4

On remarque que le coût après optimisation est bien inférieur que le coût d'avant. D'où on a économisé 24% de prix totale de notre projet. Cette diminution aura un impact direct sur le coût des accessoires et pièce spéciale.

Conclusion :

Dans ce chapitre on conclure que notre programme qui couple l'algorithme de SCE et le logiciel EPANET nous a permis obtenus à un réseau optimale de coté économique et technique. La différence est remarquable donc notre programme est fiable et fonction bien.

CONCLUSION GENERALE

CONCLUSION GENERALE :

La conception optimale d'un réseau de distribution d'eau est une tâche complexe. Divers algorithmes de recherche ont été proposés et essayés. Les principales préoccupations sont d'obtenir la solution optimale avec un coût de conception minimal et en même temps, de satisfaire la pression requise à certains nœuds de demande et de n'utiliser que les diamètres de tuyauterie disponibles sur le marché.

L'objectif de cette étude est d'utiliser un algorithme évolutif (SCE) pour concevoir l'optimisation du réseau de distribution d'eau et déterminer les paramètres du réseau.

La conception ou la réhabilitation d'un réseau de distribution d'eau peut être effectuée manuellement ou automatiquement. La conception manuelle est évidemment très lourde et, surtout, les performances dépendent de l'expérience du modéleur.

De plus, il est très long de déterminer la solution la moins coûteuse répondant aux contraintes spécifiques de l'utilisateur. En revanche, le schéma de conception automatique permet de surmonter le processus manuel fastidieux. Le procédé automatique couple un outil d'optimisation (SCE) et un modèle de simulation de réseau hydraulique EPANET. Le détail est bien expliqué au chapitre II.

Dans chapitre I nous avons remarqué que certains algorithmes traditionnels sont incapables d'arriver à des minima globaux et nécessitent un effort de calcul considérable pour trouver même l'optimum local. En outre, ils ne peuvent pas fonctionner avec des variables discrètes. Comme le temps PC augmente de façon exponentielle avec le nombre des tubes considérés, l'application des algorithmes d'optimisation traditionnels est généralement limitée à un réseau plus petit.

Cette étude a démontré la faisabilité du Shuffled Complex Evolution (SCE) dans la conception de réseau de distribution d'eau. Comme on l'a vu dans chapitre III, la SCE présente un équilibre délicat entre l'exploration et l'exploitation par rapport à l'AG standard. Il recherche dans différentes directions à l'intérieur de l'espace réalisable ; SCE est basé sur la méthode Nelder et Mead Simplex.

Dans ce système d'approvisionnement en eau, chaque conduite est considérée comme une variable ; il y a un total de 19 variables. Étant donné qu'il y a 17 tailles des conduites commerciales, le nombre de combinaisons possibles est inabordable pour une approche d'essai et d'erreur. La SCE n'a recherché intelligemment qu'un seul fichier un certain nombre de combinaisons pour arriver à une région optimale réalisable dans l'espace de solution. Afin d'augmenter les chances de convergence de la SCE dans la région optimale et de réduire le risque d'être piégé dans l'optima local. En conséquence, la solution du tirage final est nettement plus optimale.

Une demande plus élevée. SCE produit un ensemble de solutions réalisables dans la génération finale qui est très utile au décideur pour choisir une solution appropriée d'intérêt basé sur le budget et d'autres critères de conception.

Dans le chapitre III Les résultats de l'étude montrent que l'utilisation de notre programme d'optimisation sur le réseau permet d'obtenir un coût total minimisé de 24% par rapport au coût précédent. Il donne un réseau hydrauliquement amélioré et fiable.

REFERENCES
BIBLIOGRAPHIQUES

Recherche bibliographique :

- [1] Zong Woo Geem, Joong Hoon Kim, et G. V. Loganathan, « Un nouvel algorithme d'optimisation heuristique: recherche Harmony », *SIMULATION*, vol. 76, n° 2, p. 60-68, févr. 2001.
- [2] H. D. Sherali, E. P. Smith, et S. Kim, « A Pipe Reliability and Cost Model for an Integrated Approach Toward Designing Water Distribution Systems », in *Global Optimization in Engineering Design*, I. E. Grossmann, Éd. Boston, MA: Springer US, 1996, p. 333-354.
- [3] E. Alperovits et U. Shamir, « Design of optimal water distribution systems », *Water Resour. Res.*, vol. 13, n° 6, p. 885-900, déc. 1977.
- [4] A. Tits, A. Wächter, S. Bakhtiari, T. Urban, et C. Lawrence, « A Primal-Dual Interior-Point Method for Nonlinear Programming with Strong Global and Local Convergence Properties », *SIAM J. Optim.*, vol. 14, n° 1, p. 173-199, janv. 2003.
- [5] T. D. Prasad, S.-H. Hong, et N. Park, « Reliability based design of water distribution networks using multi-objective genetic algorithms », *KSCE J. Civ. Eng.*, vol. 7, n° 3, p. 351-361, mai 2003.
- [6] A. R. Simpson, G. C. Dandy, et L. J. Murphy, « Genetic Algorithms Compared to Other Techniques for Pipe Optimization », *J. Water Resour. Plan. Manag.*, vol. 120, n° 4, p. 423-443.
- [7] « (2) (PDF) Genetic Algorithms for Least-Cost Design of Water Distribution Networks », *ResearchGate*. [En ligne]. Disponible sur: https://www.researchgate.net/publication/235694770_Genetic_Algorithms_for_Least-Cost_Design_of_Water_Distribution_Networks.
- [8] « (2) Genetic algorithm for optimization of water distribution systems | Request PDF », *ResearchGate*. [En ligne]. Disponible sur: https://www.researchgate.net/publication/223126345_Genetic_algorithm_for_optimization_of_water_distribution_systems..
- [9] « Application de l'optimisation globale à la conception des réseaux de canalisations », in *Proc. Int. Conf. Hydroinformatique-98*, p. 989-996.
- [10] Duan, Q., Sorooshian, S., and Gupta, « Optimal use of the SCE-UA global optimization method for calibrating watershed models », *J. Hydrol.*, vol. 158, n° 3-4, p. 265-284, juin 1994.
- [11] « (PDF) Algorithmes génétiques comparés à d'autres techniques d'optimisation de tuyaux | Angus Simpson et Graeme Dandy - Academia.edu ». [En ligne]. Disponible sur: https://www.academia.edu/15787278/Genetic_Algorithms_Compared_to_Other_Techniques_for_Pipe_Optimization. [Consulté le: 22-juin-2019].
- [12] « S.J. Van Vuuren, P.G. Van Rooyen, J.E. Van Zyl, M. Van Dijk, Application and Conceptual Development of Genetic Algorithms for Optimisation, Water Resources and Environmental Engineering I: Surface and Groundwater - Google Livres ». [En ligne].
- [13] E. Reehuis, Multiobjective Robust Optimization of Water Distribution Networks, A Masters thesis submitted to Leiden Institute of Advanced Computer Science (LIACS), Leiden University, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands, 2010.
- [14] Vasques, J.A.; Mailer, H.R.; Lence, B.J.; Tolson, B.A.; Foschi, R.O. Achieving water quality system reliability using genetic algorithms. *J. Environ. Eng.* **2000**, *126*, 954-962
- [15] Cooper, V.A.; Nguyen, V.T.V.; Nicell, J.A. Evaluation of global optimization methods for conceptual rainfall-runoff model calibration. *Water Sci. Technol.* **1997**, *36*, 53-60.
- [16] Franchini, M.; Galeati, G.; Berra, S. Global optimization techniques for the calibration of conceptual rainfall-runoff models. *Hydrol. Sci. J.* **1998**, *43*, 443-458.

- [17] Kuczera, G. Efficient subspace probabilistic parameter optimization for catchment models. *Water Resour. Res.* **1997**, *33*, 177–185.
- [18] Thyer, M.; Kuczera, G.; Bates, B.C. Probabilistic optimization for conceptual rainfall-runoff models: a comparison of the shuffled complex evolution and simulated annealing algorithms. *Water Resour. Res.* **1999**, *35*, 767–773.
- [19] Ngo, L.L.; Madsen, H.; Rosbjerg, D. Simulation and optimization modelling approach for operation of the Hoa Binh reservoir, Vietnam. *J. Hydrol.* **2007**, *336*, 269–281.
- [20] Kerachian, R.; Karamouz, M. A stochastic conflict resolution model for water quality management in reservoir-river systems. *Adv. Water Resour.* **2007**, *30*, 866–882.
- [21] Tolson, B.A.; Shoemaker, C.A. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resour. Res.* **2007**, *43*, doi:10.1029/2005WR004723.
- [22] Duan, Q.; Sorooshian, S.; Gupta, V.K. Effective and efficient global optimization for conceptual rainfall-runoff models. *Water Resour. Res.* **1992**, *28*, 1015–1031.
- [23] Holland, J.H. *Adaptation in Natural and Artificial System*; University of Michigan Press: Ann Arbor, MI, USA, 1975.
- [24] A. R. Simpson, G. C. Dandy, et L. J. Murphy, « Genetic Algorithms Compared to Other Techniques for Pipe Optimization », *J. Water Resour. Plan. Manag.*, vol. 120, n° 4, p. 423-443.
- [25] S. Chandniha, « Optimal design of water distribution network using shuffled complex evolution », *J. Inst. Eng.*
- [26] search results, *Algorithmes génétiques pour la recherche, l'optimisation et l'apprentissage automatique*, 1 edition. Boston: Addison-Wesley Professional, 1989.
- [27] « (PDF) Algorithmes génétiques comparés à d'autres techniques d'optimisation de tuyaux | Angus Simpson et Graeme Dandy - Academia.edu ». [En ligne]. Disponible sur: https://www.academia.edu/15787278/Genetic_Algorithms_Compared_to_Other_Techniques_for_Pipe_Optimization. [Consulté le: 23-mai-2019].
- [28] « (1) (PDF) A Shuffled Complex Evolution Metropolis algorithm optimization and uncertainty assessment of hydrologic models parameters », *ResearchGate*. [En ligne]. Disponible sur: https://www.researchgate.net/publication/228696681_A_Shuffled_Complex_Evolution_Metropolis_algorithm_optimization_and_uncertainty_assessment_of_hydrologic_models_parameters. [Consulté le: 21-avr-2019].
- [29] Roland Yonaba, « Introduction au modèle EPANET », 09:38:33 UTC.
- [30] « EPANET | Water Research | US EPA ». [En ligne]. Disponible sur: <https://www.epa.gov/water-research/epanet>. [Consulté le: 20-avr-2019].
- [31] 2. Duan, Q., Sorooshian, S., and Gupta, « Optimal use of the SCE-UA global optimization method for calibrating watershed models », *J. Hydrol.*, vol. 158, n° 3-4, p. 265-284, juin 1994.
- [32] « (11) (PDF) A Shuffled Complex Evolution Approach for Effective and Efficient Global Minimization », *ResearchGate*. [En ligne]. Disponible sur: https://www.researchgate.net/publication/226261472_A_Shuffled_Complex_Evolution_Approach_for_Effective_and_Efficient_Global_Minimization. [Consulté le: 23-mai-2019].
- [33] « (1) (PDF) A Shuffled Complex Evolution Metropolis algorithm optimization and uncertainty assessment of hydrologic models parameters », *ResearchGate*. [En ligne]. Disponible sur: https://www.researchgate.net/publication/228696681_A_Shuffled_Complex_Evolution_Metropolis_algorithm_optimization_and_uncertainty_assessment_of_hydrologic_models_parameters/figures. [Consulté le: 21-avr-2019].
- [34] « Cours de Python », *Bestcours*. [En ligne]. Disponible sur: <https://www.bestcours.com/programmation/python/739-cours-de-python-pdf.html>. [Consulté le: 14-mai-2019].
- [35] « Learn Python (Programming Tutorial for Beginners) ». [En ligne]. Disponible sur:

- <https://www.programiz.com/python-programming#tutorial>. [Consulté le: 19-mai-2019].
- [36] « Introduction à Python ». [En ligne]. Disponible sur: https://www.w3schools.com/python/python_intro.asp. [Consulté le: 19-mai-2019].
- [37] « Learn Python (Programming Tutorial for Beginners) ». [En ligne]. Disponible sur: <https://www.programiz.com/python-programming>. [Consulté le: 21-avr-2019].
- [38] « Python Tutorial: Learn Python For Free », *Codecademy*. [En ligne]. Disponible sur: <https://www.codecademy.com/learn/learn-python>. [Consulté le: 19-mai-2019].
- [39] « Learn Python (Programming Tutorial for Beginners) ». [En ligne]. Disponible sur: <https://www.programiz.com/python-programming#tutorial>. [Consulté le: 19-mai-2019].
- [40] « Learn Python (Programming Tutorial for Beginners) ». [En ligne]. Disponible sur: <https://www.programiz.com/python-programming>. [Consulté le: 21-avr-2019].
- [41] « Project Jupyter ». [En ligne]. Disponible sur: <https://www.jupyter.org>. [Consulté le: 19-mai-2019].
- [42] « Comment installer Anaconda et Python pour faire du Machine Learning », *Mr. Mint : Apprendre le Machine Learning de A à Z*, 20-sept-2017. .
- [43] « Introduction to Visual Studio », *GeeksforGeeks*, 30-nov-2018.

ANNEXES

ANNEXE V.1: Prix et épaisseurs des conduites.

UNITE TUBE

GAMME DE PRIX TUBES PEHD EAU (PE80 & PE100) (Prix Valable à compter du : 01 AVRIL 2010) Selon la Norme ISO 4427

TUBES PEHD : PN 06					
CODE	Diam Ext	EP(mm)	Prix HT (DA/ML)	Prix TTC (DA/ML)	
-	Ø20	-			PE80
P3513001	Ø25	-	38,22	42,48	PE80
P3513006	Ø32	-	48,28	68,87	PE80
P3513009	Ø40	2	94,6	78,88	PE80
P3513012	Ø50	2,4	98,25	114,95	PE80
P3513015	Ø63	3	100,25	137,49	PE80
P3513018	Ø75	3,6	225,21	263,49	PE80
P3513022	Ø90	4,3	296,03	346,36	PE100
P3513025	Ø110	4,9	357,86	418,7	PE100
P3513040	Ø125	-	473,08	553,5	PE100
P3513030	Ø160	5,8	770,35	901,89	PE100
P3513044	Ø200	7,2	1163,39	1361,17	PE100
P3513059	Ø250	9,1	1950	2231,5	PE100
P3513042	Ø315	11,4	3080	3603,6	PE100
P3513039	Ø400	14,5	5134,24	6007,06	PE100
P3513051	Ø500	19,1	6938,13	8117,61	PE100
P3513052	Ø630	24,1	12682,5	14038,83	PE100

TUBES PEHD : PN 10					
CODE	Diam Ext	EP(mm)	Prix HT (DA/ML)	Prix TTC (DA/ML)	
P3513000	Ø20		28,07	34,01	PE80
P3513004	Ø25	2,5	38,28	44,79	PE80
P3513007	Ø32	2,5	55,48	64,91	PE80
P3513010	Ø40	3	94,1	98,48	PE80
P3513013	Ø50	3,7	138,26	158,28	PE80
P3513016	Ø63	4,7	209,68	245,34	PE80
P3513019	Ø75	5,6	300,5	351,59	PE80
P3513023	Ø90	5,4	359,09	420,14	PE100
P3513026	Ø110	6,6	540,5	632,39	PE100
P3513028	Ø125	7,4	680,13	772,36	PE100
P3513031	Ø160	9,5	1142,25	1336,43	PE100
P3513033	Ø200	11,9	1650,84	1931,48	PE100
P3513035	Ø250	14,8	2590	3036,15	PE100
P3513041	Ø315	18,7	4093,57	4789,47	PE100
P3513038	Ø400	23,7	6098,74	7837,53	PE100
P3513049	Ø500	29,7	10152,51	11878,44	PE100
P3513048	Ø630	37,7	15733,76	18411,38	PE100

TUBES PEHD : PN 16					
CODE	Diam Ext	EP(m m)	Prix HT (DA/ML)	Prix TTC (DA/ML)	
P3513002	Ø20	2,3	32,12	37,65	PE80
P3513003	Ø25	2,8	50,65	58,49	PE80
P3513005	Ø32	3,6	78,51	91,88	PE80
P3513008	Ø40	4,5	120,95	141,51	PE80
P3513011	Ø50	5,6	188,28	220,29	PE80
P3513014	Ø63	7,1	297,95	348,5	PE80
P3513017	Ø75	8,4	407,8	477,13	PE80
P3513020	Ø90	8,2	610,71	597,53	PE100
P3513021	Ø110	10	767,19	888,91	PE100
P3513024	Ø125	11,4	967	1132,44	PE100
P3513027	Ø160	14,6	1550	1828,3	PE100
P3513029	Ø200	18,2	2472	2892,94	PE100
P3513032	Ø250	22,7	3915	4463,55	PE100
P3513034	Ø315	28,6	5850	6844,5	PE100
P3513036	Ø400	36,3	9320	10904,4	PE100
P3513037	Ø500	45,4	14911	17445,87	PE100
P3513043	Ø630	57,2	23123,5	27054,5	PE100

TUBES PEHD : PN 20					
CODE	Diam Ext	EP(mm)	Prix HT (DA/ML)	Prix TTC (DA/ML)	
P3513101	Ø20	2,3	38,01	44,47	PE80
P3513102	Ø25	3	53,29	62,35	PE100
P3513103	Ø32	3,6	82,22	96,2	PE100
P3513104	Ø40	4,5	128,15	149,94	PE100
P3513105	Ø50	5,6	200,7	234,82	PE100
P3513106	Ø63	7,1	338,73	396,31	PE100
P3513107	Ø75	8,4	447,36	623,4	PE100
P3513108	Ø90	10,1	637,09	733,7	PE100
P3513109	Ø110	12,3	948	1110,17	PE100
P3513110	Ø125	14	1202,75	1407,22	PE100
P3513111	Ø160	17,9	1947,58	2278,88	PE100
P3513112	Ø200	22,4	3121,79	3952,3	PE100
P3513113	Ø250	27,9	4846,33	5579,21	PE100
P3513114	Ø315	35,2	7588,88	8875,48	PE100
P3513115	Ø400	44,7	12110,03	14158,74	PE100
P3513054	Ø500	55,8	19053,81	22304,68	PE100
-	Ø630	-			PE100