

**Sujet :**

**Développement d'un Outil d'Aide  
À la Planification du Réseau Cœur GSM**

Elaboré par :

**Mohamed Raâfat OMRI**

**RAPPORT DE  
PROJET DE FIN D'ETUDES**

Présenté en vue de l'obtention de la  
Licence Appliquée en Sciences et Techniques  
de l'Information et de Communication

UNIVERSITE VIRTUELLE DE TUNIS

**Année universitaire : 2010/2011**

Je tiens à remercier particulièrement Mr. Saber BOUAZZI, mon encadreur **de la société d'accueil pour ses directives et conseils pertinents afin de mener à terme ce travail.**

Je remercie également Mr. Kamel KHEDHIRI mon encadreur de **l'Université Virtuelle de Tunis pour son soutien et orientation.**

Mes remerciements vont également à Monsieur le Président et les **membres du jury qui acceptent d'évaluer mon travail.**

<b>Introduction générale</b> .....	1
<b>Chapitre 1 : Architecture du réseau GSM</b> .....	2
1.1 Introduction.....	2
1.2 La station mobile (le téléphone portable).....	3
1.3 Le sous système Radio (BSS).....	3
1.4 Le sous système Réseau (NSS).....	4
1.5 Le centre d'exploitation et de maintenance OMC.....	4
<b>Chapitre 2 : Généralités sur la planification des réseaux</b> .....	5
2.1 Planification cellulaire : généralités.....	5
2.2 Dimensionnement des réseaux cellulaires.....	6
2.3 Planification du sous système Radio.....	6
2.4 Planification de la partie fixe (NSS).....	7
2.4.1 Présentation générale et objectifs.....	7
2.4.2 Données d'entrée et contraintes.....	7
2.5 Conclusion.....	8
<b>Chapitre 3 : Formulation des problématiques de planification du NSS</b> .....	9
3.1 Localisation des concentrateurs.....	9
3.1.1 Introduction.....	9
3.1.2 Formulation du problème.....	10
3.2. Affectation des terminaux aux concentrateurs avec prise en charge de la signalisation.....	12
3.2.1 Introduction.....	12
3.2.2 Formulation du problème.....	12
3.3 Planification des zones de localisation (Local Area).....	14
3.3.1 Introduction au problème.....	14
3.3.2 Formulation du problème.....	15
3.4 Conclusion.....	16
<b>Chapitre 4 : Les algorithmes heuristiques : principe et description</b> .....	17
4.1 Introduction.....	17
4.2 Méthode de recherche Tabou.....	17
4.3 Les algorithmes génétiques (AG).....	18
4.4 Le recuit simulé (RS).....	19
4.4.1 Principe générale.....	19
4.4.2 Le paramètre température.....	20

4.5 Comparaison des différents algorithmes.....	20
<b>Chapitre 5 : Résolution de la problématique de localisation des concentrateurs par la méthode de recherche Tabou.....</b>	<b>21</b>
5.1 Introduction.....	21
5.2 Plateforme informatique.....	21
5.2.1 Choix de l’outil de programmation.....	21
5.2.2 Les modules utilisés dans l’application.....	21
5.2.3 La solution proposée par l’application.....	24
5.3 Conclusion.....	24
<b>Conclusion générale.....</b>	<b>25</b>
<b>Bibliographie.....</b>	<b>26</b>
<b>ANNEXES.....</b>	<b>27</b>
ANNEXE A.....	27
ANNEXE B.....	29
ANNEXE C.....	35

# Introduction générale

La concurrence dans le marché des télécommunications mobiles devient de plus en plus agressive et les opérateurs doivent faire face à cette rivalité en réduisant au maximum le coût à déployer dans le réseau tout en assurant les demandes en trafic et en couverture des utilisateurs qui ne cessent d'augmenter.

Au début de la vie d'un réseau mobile, la partie radio coûte trop chère comparée à la partie réseau, mais au fur et à mesure que le réseau s'élargie, la partie réseau devient aussi coûteuse et donc elle doit faire l'objet d'une planification dès le départ.

Ce projet de fin d'étude s'intéresse à la planification de la partie fixe du réseau GSM (Le NSS) et suggère la résolution du problème de localisation des sites d'installation des BSCs et MSCs. Après une formulation de ce problème, il s'avère qu'il ne peut pas être résolu avec des méthodes exactes, ce qui oblige l'utilisation de l'une des méthodes qui donne des solutions proches de l'optimale, la méthode choisie est Recherche Tabou qui est simple à mettre en œuvre et permet essentiellement d'échapper du problème des minimums locaux. Ce problème est implanté dans une application Java qui permet d'afficher graphiquement une architecture globale du réseau après la saisie des données d'entrées qui sont collectées lors de la planification de la partie radio.

Pour introduire le sujet, le lecteur trouvera dans le premier chapitre une brève description de l'architecture du réseau GSM et de ces composants. Le deuxième chapitre discutera les étapes de planification d'un réseau cellulaire parmi lesquelles on trouve celle du réseau fixe. Le troisième chapitre sera consacré à la description et à la formulation des différentes problématiques qui se posent lors de la planification du réseau fixe. Et avant de se lancer dans le chapitre cinq qui explique les différents modules utilisés dans l'application, le chapitre quatre explique le principe de quelques méthodes heuristiques utilisées pour la résolution des problèmes tel que celui étudié dans ce travail.

# Chapitre 1 : Architecture du réseau GSM

Le présent chapitre décrit les différents composants du réseau GSM, et cela pour faciliter au lecteur la compréhension des autres chapitres.

## 1.1 Introduction

Le réseau GSM peut être divisé en 3 parties :

1. La station mobile qui est transportée par l'utilisateur.
2. Le sous système radio (BSS : Base Station Sub-system) qui contrôle les liaisons radio qui s'établissent avec le téléphone portable.
3. Le sous système réseau (NSS : Network Sub-system) qui permet la connexion d'un mobile vers un autre mobile ou vers un utilisateur du réseau fixe.

Ainsi, on peut représenter schématiquement un réseau radiomobile de la manière suivante :

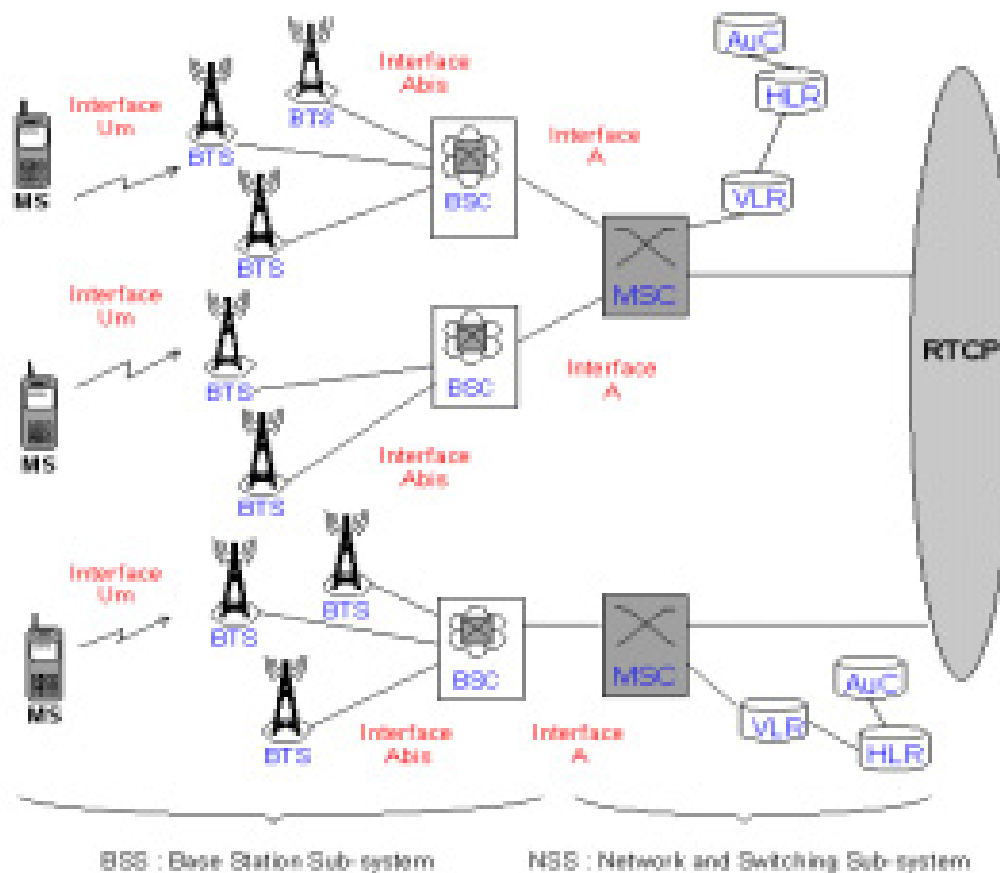


Figure 1 architecture du réseau GSM

## 1.2. La station mobile (le téléphone portable)

La station mobile est composée de l'équipement possédant son identité internationale IMEI (International Mobile Equipment Identity) et de la carte SIM (Subscriber Identity Mobile) contenant principalement l'identité de l'abonné IMSI (International Mobile Subscriber Identity) et la clé secrète pour la sécurité Ki (Individual Subscriber Authentication Key) servant à l'authentification et au chiffrement de la liaison radio avec la station de base.

Le MSRN ou Mobile Station Roaming Number est un numéro de téléphone défini par les recommandations E.164 afin de router les appels téléphoniques dans un réseau mobile, depuis un Gateway Mobile Switching Center (GMSC) jusqu'à la cible Mobile service Switching Center (MSC).

## 1.3. Le sous système radio (BSS)

Le sous système radio est composé de deux parties : la station de base (BTS : Base Transceiver Station) et du contrôleur de station de base (BSC : Base Station Controller). Ces deux parties communiquent entre elles par l'intermédiaire de l'interface Abis (voir figure 1).

Le BTS gère une cellule, il définit entre autre la taille de la cellule suivant l'environnement. Ainsi dans une zone urbaine, on déploiera plus de BTS que dans une zone rurale. En effet, les BTS sont résistants, fiables, mobiles et peu onéreux.

Le BTS assure également les transmissions radio entre les mobiles et le réseau, ainsi il gère :

- L'émission/réception radio.
- La couche physique (émission en TDMA, saut de fréquence lent).
- Les mesures de qualité des signaux reçus.
- La liaison radio.

Toute station de base est surveillée par un poste appelé BSC. Un réseau GSM comporte plusieurs BSC, qui commandent chacun une ou plusieurs stations de base. Le BSC est responsable de toutes les fonctions liées à la transmission radio, comme le handover, la gestion des ressources du réseau, le niveau de puissance des BTS et des mobiles connectés ...

## 1.4. Le sous système réseau (NSS)

L'élément central du sous système réseau est le MSC. Il agit comme un commutateur classique de réseau numérique (RNIS) auquel on a ajouté les fonctionnalités nécessaires pour gérer la mobilité des abonnés, comme l'enregistrement, l'authentification, la mise à jour de la localisation, les handovers et le routage des appels. A chacune de ces fonctionnalités correspond une entité fonctionnelle, lesquelles forment avec le commutateur classique, le sous système réseau.

Quatre bases de données sont associées au MSC : le HLR, le VLR, le EIR, et le AuC :

- Le HLR contient toutes les caractéristiques d'abonnement de tous les utilisateurs du réseau GSM, leurs identités IMSI et MSISDN (numéro d'appel du portable) ainsi que les localisations des portables. Le HLR travaille en étroite collaboration avec les différents VLR, notamment pour les handovers, et la numérotation.
- Le VLR contient les informations nécessaires à la gestion des mobiles présents dans sa zone notamment celles nécessaires à la numérotation, la localisation, et le type d'abonnement. Elle gère plusieurs dizaines de milliers d'abonnés, ce qui correspond à une région.
- Le EIR contient une liste de tous les mobiles valides sur le réseau, et chaque téléphone portable est identifié dans cette base de données par un numéro IMEI (International Mobile Equipment Identity). L'IMEI est marqué comme invalide si le mobile a été déclaré comme volé.
- La base de données AuC (Authentication Center), est une base de protection qui contient une copie d'une clé secrète, également contenue dans la carte SIM de chaque abonné. Cette dernière est utilisée pour l'authentification d'un portable. Ce contrôle se fait par l'intermédiaire de canaux radio.

## 1.5 Le centre d'exploitation et de maintenance OMC

Le centre d'exploitation et de maintenance OMC (Operation and Maintenance Center) est décomposé en deux centres l'OMC-R et l'OMC-S.

Le centre d'exploitation et de maintenance du sous-système réseau OMC-S (Operation and Maintenance Center Switching Part) supervise, détecte et corrige les anomalies du NSS.

Le centre d'exploitation et de maintenance OMC-R (Operation and Maintenance Center-Radio part) exploite et maintient le sous-système radio.



# Chapitre 2 : Généralités sur la planification des réseaux cellulaires

Nous aborderons dans la première partie de ce chapitre quelques généralités liées à la planification cellulaire, nous parlerons ensuite des étapes de dimensionnement, de planification de la partie radio et de planification de la partie réseau.

## 2.1 Planification cellulaire : généralités

La planification d'un réseau cellulaire est une tâche difficile et continue dont le résultat conditionne la réussite de l'opérateur. Celui-ci doit répondre aux besoins des usagers en garantissant une qualité de service acceptable tout en minimisant les coûts d'investissement. En plus, l'opérateur doit être toujours à l'écoute des différents changements et événements qui se déroulent sur son réseau afin de s'adapter aux besoins de ces clients.

Ainsi, l'opérateur doit déterminer un calendrier de modification de son réseau permettant de supporter les évolutions futures du trafic et des technologies en garantissant la qualité attendue des communications et en induisant le coût le plus faible possible.

Pour maîtriser le problème de planification, l'opérateur doit évaluer en premier lieu les caractéristiques de l'environnement à couvrir, à savoir les caractéristiques radio de l'environnement de propagation, puis les caractéristiques des abonnés à desservir qui se résument dans leur densité, mobilité, et leurs besoins en trafic. Ainsi, en possédant en main des modèles de propagation appropriés et des prévisions en trafic sur le court et le long terme, en réalisant des mesures sur le réseau existant, l'opérateur doit aboutir à un ensemble de décisions en établissant :

- Un plan de stations de bases qui va déterminer leurs emplacements, leurs capacités et les puissances mises en jeu.
- Un plan de fréquences affectant à chaque station de base les fréquences de travail.
- Un plan des équipements du réseau fixe.
- Un réseau de connexion entre les différentes entités du réseau.

Le processus de planification se déroule suivant les étapes suivantes :

- Dimensionnement des équipements du réseau en déterminant le volume des équipements et logiciels à acquérir et à déployer pour la fourniture des services de télécommunication mobiles.
- Planification du sous-système radio qui doit mener à la détermination des positions et capacités des stations de base et aussi à spécifier un plan de fréquences.
- Planification du sous-système réseau en spécifiant les emplacements et capacités des équipements du réseau fixe et les interconnexions entre eux.

## **2.2 Dimensionnement des réseaux cellulaires**

Le dimensionnement des équipements et interfaces est la première étape intervenant dans le cycle de vie d'un réseau cellulaire. Elle permet en effet de prédire, d'un premier coup, le nombre des équipements et autres moyens à acquérir pour mettre les services de télécommunications mobiles des réseaux cellulaires en marche.

Le dimensionnement des ressources du système est lié au besoin en trafic des utilisateurs. Ainsi en connaissant le trafic qui doit être écoulé par un système on peut le dimensionner. Pour le cas des réseaux cellulaires, la détermination de la charge de trafic pour chaque cellule permet de dimensionner les différents liens (trafic et signalisation) sur l'interface radio. Ensuite, le dimensionnement des BTS peut se faire à partir de la charge estimée de sa zone de couverture. Le processus se déroule ainsi de proche en proche pour dimensionner à la fin les interfaces et équipements du réseau fixe (commutateurs et bases de données).

Les trafics qu'il faut tenir en compte sont principalement :

- Le taux d'appels (nombre et durée des appels)
- Le trafic lié à la mobilité (mise à jour de localisation, handover, etc.).

## **2.3 Planification du sous système radio**

La planification du BSS comprend les étapes suivantes :

- Un plan de couverture : qui a comme objectif de déterminer les emplacements des stations de base, leurs capacités et leurs puissances.
- Un plan de fréquence associé : La réutilisation plus serrée signifie plus de capacité dans le réseau, mais plus d'interférence. Il y a un compromis entre la capacité et les interférences. L'objectif de la planification des fréquences doit réduire au minimum possible les interférences à un niveau de capacité acceptable.

- Un plan de transmission : Les dépenses pour la transmission sont de l'ordre de 20% des dépenses totales d'exploitation d'un opérateur par an. Le coût financier étant important, il faut trouver la solution la plus efficace et la moins coûteuse pour une topologie donnée du réseau.

Nous arrivons maintenant au vif de notre sujet qui est la planification du sous système réseau. Le paragraphe suivant étale les différentes problématiques liées à la planification du NSS qui vont être traitées en détail dans le chapitre 4.

## 2.4 Planification de la partie fixe (NSS)

### 2.4.1 Présentation générale et objectifs

La planification du sous-système réseau vient après celle du sous-système radio. Elle comprend les étapes suivantes:

- Détermination des nombres et localisations des concentrateurs (BSC et MSC)
- Allocation des BTS aux BSC et des BSC aux MSC.
- Définition des zones de localisation.
- Définition des connexions entre les MSC

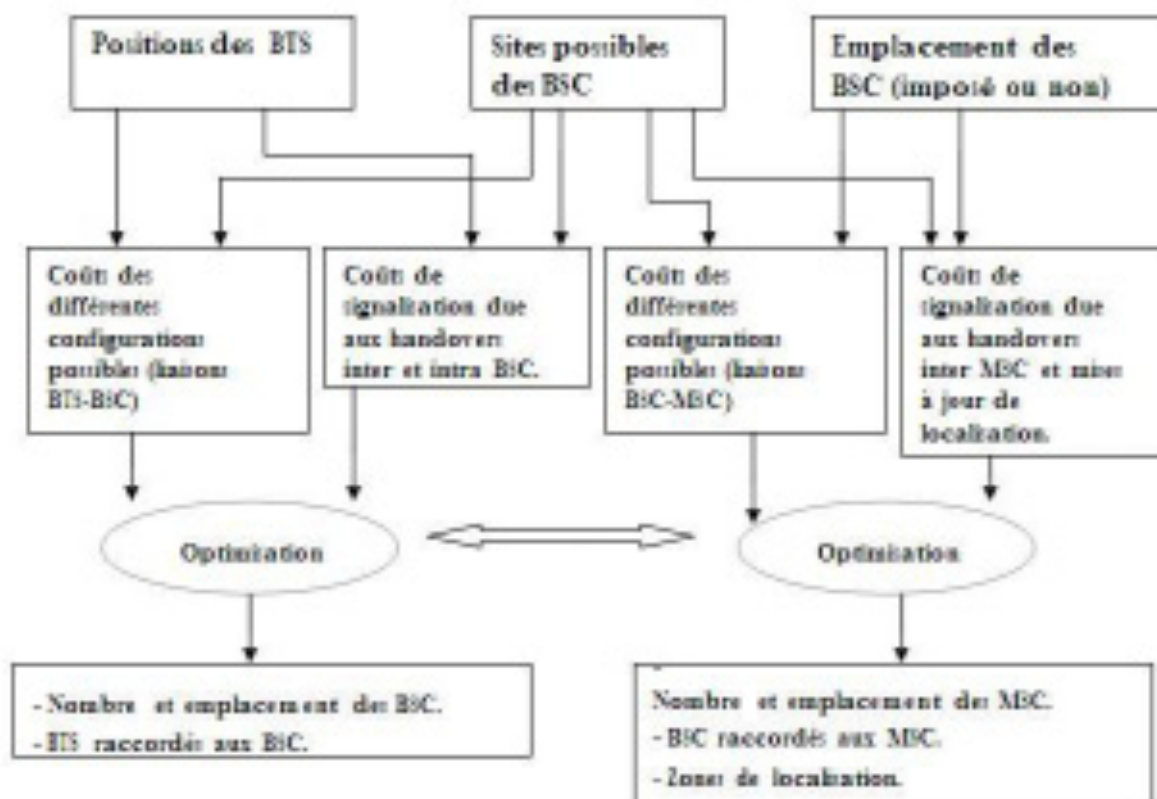


Figure 2 : Déroulement du processus de planification du réseau fixe

## 2.4.2 Données d'entrée et contraintes

L'architecture du sous-système réseau comprend les concentrateurs (BSC, MSC), les équipements tels que les bases de données, les serveurs de messagerie, les plates formes du réseau intelligent, les passerelles vers les autres réseaux ainsi que les liaisons de transmission connectant les équipements entre eux et avec les équipements des réseaux extérieurs.

Pour cela les données suivantes doivent être connues :

- Sites et capacités des stations de base.
- Capacités des différents types de concentrateurs (BSC, MSC).
- Emplacements possibles des concentrateurs.
- Coûts d'acquisition ou de location des sites des concentrateurs.
- Coûts d'exploitation et de maintenance des concentrateurs pour chaque site possible.
- Coût des liaisons de transmission entre les entités du réseau mobile.

Plusieurs contraintes sont prises en compte dans le processus de planification. Parmi les contraintes techniques, on note:

- Positions des concentrateurs qui sont souvent imposées en fonction de la disponibilité des sites.
- Capacité des liaisons de transmission entre les différents sites.
- Capacité de raccordement des concentrateurs. Cette capacité dépend des types et constructeurs de ces concentrateurs.
- Minimisation de la charge de signalisation. La signalisation ici est celle due aux différents types de handovers et de mises à jour de localisation.

Et parmi les contraintes de coût on note :

- Minimisation des coûts d'investissement, d'installation et d'exploitation des concentrateurs.
- Minimisation des coûts des liaisons de transmission.
- Minimisation des interconnexions.

## 2.5 Conclusion

Nous avons proposé dans ce chapitre quelques notions sur la planification du réseau GSM à savoir la phase de dimensionnement et celles de la planification de la partie radio et réseau. Chaque étape doit être réalisée avant le commencement de l'étape suivante.

## **Chapitre 3 : Formulation des problématiques de planification du NSS**

Dans le chapitre précédent, nous faisons une introduction à la planification de la partie fixe. Dans le présent chapitre on essaye de définir les différentes problématiques qui découlent de cette planification. On donne une formulation mathématique de chacune d'elles. La formulation du problème de localisation des concentrateurs va nous aider par la suite (chapitre 5) à concevoir un outil java qui se base sur une méthode heuristique (voir chapitre 4).

### **3.1 Localisation des concentrateurs**

#### **3.1.1 Introduction**

La première problématique à laquelle doit répondre le planificateur du sous-système réseau est de déterminer l'emplacement des concentrateurs. Le principal objectif considéré consiste à minimiser les coûts incluant coûts des liaisons de transmission, des concentrateurs, d'installation (acquisition et aménagement du site), d'exploitation et de maintenance. Le recours à des contrôleurs de capacité importantes permet de réduire leur nombre et par conséquent permet de réduire le coût des sites et de maintenance mais conduit à un surcoût au niveau des liaisons de transmission puisqu'il faut raccorder des sites terminaux éloignés. Il est donc important de définir une topologie de réseau permettant de minimiser le coût total incluant le coût des concentrateurs et celui des liaisons de transmission. Ce problème est classé NP-difficile (voir annexe A).

Les concentrateurs sont de deux types et se situent à des niveaux différents dans le réseau : Les contrôleurs de stations de base constituent des concentrateurs pour les stations de base et les commutateurs constituent à leur tour des concentrateurs pour les contrôleurs de stations de base.

Dans les paragraphes qui suivent, on donnera une formulation du problème de positionnement des concentrateurs pour donner ensuite une approche de résolution du problème par la méthode de recherche tabou que nous allons expliquer le principe dans le chapitre suivant. Reste à noter que nous désignons par concentrateur (BSC ou MSC) l'entité à laquelle des sites terminaux (BTS ou BSC) sont raccordés.

### 3.1.2 Formulation du problème

La résolution du problème de localisation des concentrateurs revient à trouver un compromis entre les coûts d'investissement des concentrateurs et les coûts des liaisons de transmission. On propose de formuler ce problème pour les deux cas suivants : localisation des MSC connue (imposé le plus souvent par des contraintes de disponibilité du site) ou non connue.

#### 3.1.2.1 Localisation des MSC supposée fixée

Le problème revient seulement à trouver les positions des sites d'installation des BSC. Le réseau à minimiser comporte un MSC et N stations de base connectées à M contrôleurs de stations de base. Chaque station de base ne peut être connectée qu'à un BSC et le nombre de BTS connectées à un BSC est limité (limite imposée par le matériel) par MAX\_BTS. Tous les BSC sont connectés à un seul MSC.

Les emplacements des BTS et du MSC sont connus, l'objectif du processus d'optimisation consiste à trouver le site optimal pour chaque BSC et à déterminer les BTS qui doivent lui être raccordées.

On a plusieurs sites possibles d'implantation des BSC, pour chaque site j on a un coût d'implantation  $F_j$  d'un concentrateur sur ce site. Le but est donc de minimiser la somme des coûts d'implantation et la somme des coûts des liaisons. Ces liaisons comportent les liaisons entre les BTS et BSC et les liaisons entre les BSC et MSC.

Soit donc :

$$X_{ij} = \begin{cases} 1 & \text{si la BTS } i \text{ est raccordé au site } j. \\ 0 & \text{sinon.} \end{cases}$$

Et :

$$Y_j = \begin{cases} 1 & \text{si une BSC est installée au site } j \\ 0 & \text{sinon.} \end{cases}$$

Il s'agit de minimiser la fonction coût suivante :

$$f = \sum_{i=1}^N \sum_{j=1}^M C_{ij} X_{ij} + \sum_{j=1}^M F_j y_j + \sum_{j=1}^M D_j y_j$$

Avec :

$C_{ij}$  : le coût de raccordement du BTS  $i$  au site  $j$ .

$F_j$  : le coût d'implantation d'un BSC sur le site  $j$ .

$D_j$  : le coût de raccordement du MSC au site  $j$ .

Il apparaît clairement que le premier et le troisième terme représentent les coûts de raccordement des BSC aux BTS et des BSC aux MSC, tandis que le deuxième terme représente le coût d'implantation des BSC.

Les contraintes du problème sont les suivantes :

-  $\sum_{j=1}^M X_{ij} = 1$  pour  $i=1, \dots, N$  cette contrainte signifie qu'un BTS ne peut être raccordé qu'à un seul BSC.

-  $\sum_{i=1}^N X_{ij} \leq K y_j$  pour  $j=1, \dots, M$ ,

Cette contrainte signifie que le nombre des BTS raccordés à un BSC ne peut dépasser la capacité  $K$  de raccordement de ce dernier.

- Pour  $i=1, \dots, N$  et  $j=1, \dots, M$  :  $X_{ij}$  et  $Y_j \in \{0,1\}$

On a pu ainsi formuler le problème de localisation des BSC sachant que la localisation des MSC est fixée. Ce problème est classé NP-difficile et sa résolution ne peut se faire rapidement que par les méthodes heuristiques (voir chapitre 4).

### 3.1.2.2 Localisation du MSC non fixé

Dans ce cas la localisation des MSC est non connue, donc il faut trouver les sites d'installation des MSC et ceux des BSC. Cette tâche est réalisée en deux étapes : il s'agit d'abord de localiser les BSC et déterminer les BTS raccordés aux BSC, et ensuite localiser les MSC et déterminer les BSC qui leur sont raccordés.

La formulation du problème est semblable à la précédente sauf qu'ici la connexion des BTS aux BSC et celle des BSC aux MSC se font séparément et les deux se formulent de la même manière.

On considère  $N$  sites terminaux et  $M$  sites possibles des concentrateurs. La fonction à minimiser est la suivante :

$$f = \sum_{i=1}^N \sum_{j=1}^M C_{ij} X_{ij} + \sum_{j=1}^M F_j y_j$$

Les paramètres et contraintes du problème sont les mêmes que pour le cas de localisation du MSC fixée.

## **3.2. Affectation des terminaux aux concentrateurs avec prise en charge de la signalisation**

Le problème d'affectation des cellules aux commutateurs dans les réseaux cellulaires pour mobiles peut se diviser en deux sous problèmes similaires, le premier est le problème d'affectation des BTS aux BSC et le deuxième est l'affectation des concentrateurs BSC aux commutateurs MSC. On va traiter dans ce paragraphe juste le premier problème puisque la démarche et la modélisation du deuxième problème sont similaires au premier.

### **3.2.1 Introduction**

Dans un réseau de radiocommunications avec les mobiles, la zone de couverture est, comme déjà évoquée précédemment, découpée géographiquement en cellules hexagonales affectées chacune à un ou deux commutateurs. Chaque cellule est couverte par un BTS qui échange des signaux avec les terminaux mais aussi avec le BSC qui lui est lié.

Les utilisateurs peuvent changer de cellule, donc de BSC éventuellement. L'opération qui consiste à noter le changement de cellule et effectuer les mises à jour nécessaires constitue un transfert intercellulaire ou une relève (handover). Quand la relève s'effectue entre deux BTSs reliés au même BSC, on parle de relève simple car les mises à jour à faire sont peu nombreuses. Par contre, quand la relève se déroule entre deux BTSs reliés à des BSCs différents, on parle de relève complexe puisque les mises à jour consomment plus de ressources.

Le problème d'affectation des BTSs aux BSCs est une minimisation d'une fonction coût composée des coûts de câblage (ou de connexion) et des coûts des relèves (simples et complexes) sous des contraintes de capacité des BSCs.

### **3.2.2 Formulation du problème**

Soient  $N$  BTSs à affecter à  $M$  BSCs, étant donné que la localisation des BTSs et BSCs est fixe et connue. Soient  $H_{ij}$  le coût par unité de temps d'une relève simple entre les BTS  $i$  et  $j$ , et  $H'_{ij}$  celui d'une relève complexe entre les BTSs  $i$  et  $j$  impliquant deux BSCs. Notons que  $H_{ij}$  et  $H'_{ij}$  sont proportionnels à la fréquence des relèves entre  $i$  et  $j$ . Désignons par  $C_{ik}$  le coût de la liaison entre le BTS $_i$  et le BSC $_k$  et par  $\lambda_i$  le nombre de TRX destinés à la cellule  $i$ . La capacité d'un commutateur  $k$  en nombre de TRX est notée  $M_k$ .



Soit :

$$X_{ik} = \begin{cases} 1 & \text{si la cellule } i \text{ est reliée au commutateur } k. \\ 0 & \text{sinon} \end{cases}$$

L'une des contraintes du problème est que chaque BTS doit être affecté à un et un seul BSC, ce qui se traduit par :

$$(1) \quad \sum_{k=1}^M X_{ik} = 1 \quad \text{pour } i=1,\dots,N$$

Soient  $Z_{ijk}$  et  $Y_{ij}$  définis par :

$Z_{ijk} = X_{ik}X_{jk}$  pour  $i,j=1,\dots,N$  et  $k=1,\dots,M$  et  $i \neq j$ .  $Z_{ijk}$  est égal à 1 si les cellules sont toutes deux connectées au même BSC  $k$ , sinon il est nul.

$$Y_{ij} = \sum_{k=1}^M Z_{ijk} = 1 \quad \text{pour } i,j=1,\dots,N \text{ et } i \neq j$$

$Y_{ij}$  prend la valeur 1 si les cellules  $i$  et  $j$  sont connectées à un même BSC et 0 sinon.

Ainsi, on peut exprimer le coût  $f$  par unité de temps ainsi :

$$(2) \quad f = \sum_{i=1}^N \sum_{k=1}^M C_{ik}X_{ik} + \sum_{i=1}^N \sum_{j=1, j \neq i}^N H'_{ij}(1 - Y_{ij}) + \sum_{i=1}^N \sum_{j=1, j \neq i}^N H_{ij}Y_{ij}$$

Le premier terme représente le coût de liaison, le deuxième exprime le coût des relèves complexes et le troisième celui des relèves simples. On signale ici que la fonction coût  $f$  est une fonction quadratique en  $X$  (car  $Y$  est quadratique en  $X$ ).

La contrainte sur les capacités des BSC impose la relation suivante (selon laquelle la charge totale de toutes les cellules ne dépasse par la capacité du BSC correspondant):

$$(3) \quad \sum_{i=1}^N \lambda_i X_{ik} \leq M_k \quad \text{pour tout } k=1,\dots,M$$

Avec  $\lambda_i$  le nombre d'appels par unité de temps destinés à la cellule  $i$  et  $M_k$  la capacité du BSC numéro  $k$ .

Le problème peut alors se reformuler ainsi : Minimiser (2) sous les contraintes (1) et (3).

On peut simplifier le problème en posant :  $h_{ij} = H'_{ij} - H_{ij}$ , la relation (2) peut s'écrire sous la forme :

$$f = \sum_{i=1}^N \sum_{k=1}^M C_{ik} X_{ik} + \sum_{i=1}^N \sum_{j=1, j \neq i}^N h_{ij} (1 - Y_{ij}) + \sum_{i=1}^N \sum_{j=1, j \neq i}^N H_{ij}$$

Le dernier terme étant une constante, donc le problème d'affectation des BTSs aux BSCs peut prendre la forme suivante :

Minimiser :

$$f = \sum_{i=1}^N \sum_{k=1}^M C_{ik} X_{ik} + \sum_{i=1}^N \sum_{j=1, j \neq i}^N h_{ij} (1 - Y_{ij})$$

Tel que:

$$X_{ik} = 0 \text{ ou } 1 \text{ pour } i=1, \dots, N \text{ et } k=1, \dots, M$$

$$\sum_{k=1}^M X_{ik} = 1 \text{ pour } i=1, \dots, N$$

$$\sum_{i=1}^N \lambda_i X_{ik} \leq M_k \text{ pour tout } k=1, \dots, M$$

$$Z_{ijk} \leq X_{ik} \text{ et } Z_{ijk} \leq Y_{jk}$$

$$Z_{ijk} \geq X_{ik} + X_{jk} - 1 \text{ et } Z_{ijk} \geq 0$$

### 3.3 Planification des zones de localisation (Local Area)

#### 3.3.1 Introduction au problème

Pour supporter un grand nombre d'abonnés, un opérateur radiomobile découpe le territoire à servir par des surfaces de petites tailles appelées cellules. Le terminal à l'intérieur d'une cellule communique avec le réseau mobile via une BTS qui représente le point d'accès au réseau par tous les terminaux qui résident dans cette cellule.

Lorsqu'un terminal entre dans une autre cellule, son point d'accès au réseau devient immédiatement la BTS de la nouvelle cellule. Et puisque les terminaux sont libres de passer d'une cellule à une autre, il devient nécessaire de mettre en œuvre un mécanisme qui permet l'enregistrement des emplacements des différents terminaux. Sinon, et lorsqu'un terminal est appelé, le réseau ne connaîtra pas dans quelle cellule se trouve ce terminal.

Pour localiser un terminal, l'ensemble des cellules est partitionné en des zones dites zones de localisation ou LA (Location Area), chaque LA contient un ensemble de cellules. Et ainsi, l'abonné doit enregistrer sa localisation chaque fois qu'il entre

dans une nouvelle LA et lorsqu'un appel arrive, le réseau cherche le terminal en diffusant la demande de recherche dans toutes les cellules appartenant au LA dans laquelle le terminal est enregistré.

La planification des zones de localisation est l'une des étapes primordiales dans la planification de la partie fixe d'un réseau cellulaire, il s'agit de partitionner l'ensemble des cellules en des zones de MSC/VLR.

Une zone de localisation ou LA (Location Area) est une zone contenant un ensemble de cellules dans lesquelles le système lance des messages de recherche d'un terminal mobile lors de l'établissement d'un appel, ce terminal doit donc faire des mises à jour de localisation chaque fois qu'il entre dans une nouvelle LA. Supposons que nous n'avons qu'une seule LA pour tout le système, dans ce cas il n'y a plus de mises à jour de localisation mais les messages de recherche vont être diffusés partout et donc une énorme consommation des ressources va être engendrée sans revenue pour l'opérateur. Et dans le cas contraire où chaque cellule représente une LA, les messages de recherches vont se restreindre à cette cellule mais les messages de mise à jour de localisation vont être très nombreux et leurs nombres croît surtout si la mobilité de la population est grande. Il faut donc trouver un compromis entre ces deux cas extrêmes en cherchant le meilleur partitionnement des cellules de façon à minimiser à la fois le coût des mises à jour de localisation et celui des messages de recherche ou de paging.

### 3.3.2 Formulation du problème

Le problème d'optimisation des tailles de zones de localisation peut être formulé de la façon suivante :

Etant donné les paramètres suivants :

- $f_{ij}$  est le flot de mobilité de la cellule  $i$  vers la cellule  $j$ .
- $g_{ij} = f_{ij} + f_{ji}$  flot total de mobilité entre les deux cellules  $i$  et  $j$ .
- $X_{ik} = 1$  si la cellule  $i$  appartient à la zone de localisation  $k$ .
- $N$  : nombre total de cellule dans le système.
- $t_i$  : nombre d'appels entrants dans la cellule  $i$  pendant un intervalle de temps considéré.
- $t_{max}$  : nombre maximum d'appels entrants dans une zone de localisation pendant un intervalle de temps considéré.

On cherche à minimiser la fonction coût suivante :

$$F = \sum_k \sum_{i=1}^N \sum_{j \in V(i)} f_{ji} x_{ik} (1 - x_{jk})$$

Sous les contraintes :

$$- \sum_{i=1}^N x_{ik} t_i \leq t_{\max}$$

Quelque soit  $K$  qui exprime que le maximum d'appels entrants par zone de localisation ne doit pas être dépassé

$$- \sum_k x_{ik} = 1 \quad \text{pour chaque cellule } i.$$

Pour dire que chaque cellule appartient à une et une seule zone de localisation.

Ce problème est aussi classé parmi les NP-difficile. Seules les approches heuristiques permettent d'approcher la solution optimale.

### 3.4 Conclusion

L'optimisation du sous-système réseau d'un réseau cellulaire se base principalement sur deux aspects: le coût des liaisons et d'implantation des concentrateurs, et le coût de signalisation engendré par les handovers et la gestion de la localisation.

Après une phase de planification initiale, que nous évoquons les principales étapes dans le chapitre précédent, il reste un travail d'ingénierie tout à fait important pour le planificateur. En effet, La montée en charge avec la croissance du nombre d'abonnés va nécessiter la densification du réseau, ainsi le processus de planification doit être répété à chaque évolution du réseau.

# Chapitre 4 : Les algorithmes heuristiques : principe et description

## 4.1 Introduction

Les différents problèmes liés à la planification de la partie fixe ne peuvent pas se résoudre en des temps de calcul raisonnables par des méthodes exhaustives.

En effet le nombre de solutions possibles devient trop grand dès que le nombre d'équipement dépasse une centaine par exemple, c'est pourquoi on doit avoir recours aux méthodes dites heuristiques qui donnent des solutions proches de la solution optimale en des temps raisonnables et en consommant moins de ressources.

Dans ce chapitre, on expose le principe de chacune des trois méthodes heuristiques les plus connues à savoir la méthode de recherche tabou qu'on va utiliser pour résoudre le problème d'affectation des cellules aux commutateurs (qui comprend l'affectation des BTS aux BSC et des BSC aux MSC), le recuit simulé et les algorithmes génétiques.

Et à la fin du chapitre, on donnera un tableau comparatif de ces trois méthodes.

## 4.2 Méthode de recherche tabou

La méthode de recherche tabou vise à diriger l'exploration de l'espace des solutions vers la découverte de bonnes solutions ou de solutions optimales.

Généralement, deux mécanismes sont utilisés pour guider la recherche. Le premier mécanisme vise à prévenir que la recherche ne fait pas de cycle à travers une liste tabou qui garde une trace des solutions récemment visitées lors de la recherche. Le second mécanisme utilise une ou plusieurs mémoires pour diriger la recherche dans des régions prometteuses ou vers des régions non encore explorées de l'espace des solutions.

Une procédure de recherche tabou peut être vue comme une combinaison de trois phases principales : recherche locale, intensification de la recherche dans des régions identifiées comme prometteuses de l'espace de solutions et enfin, redémarrage de la recherche dans une région de l'espace non encore explorée (diversification).

Voici un diagramme représentant le mécanisme de recherche

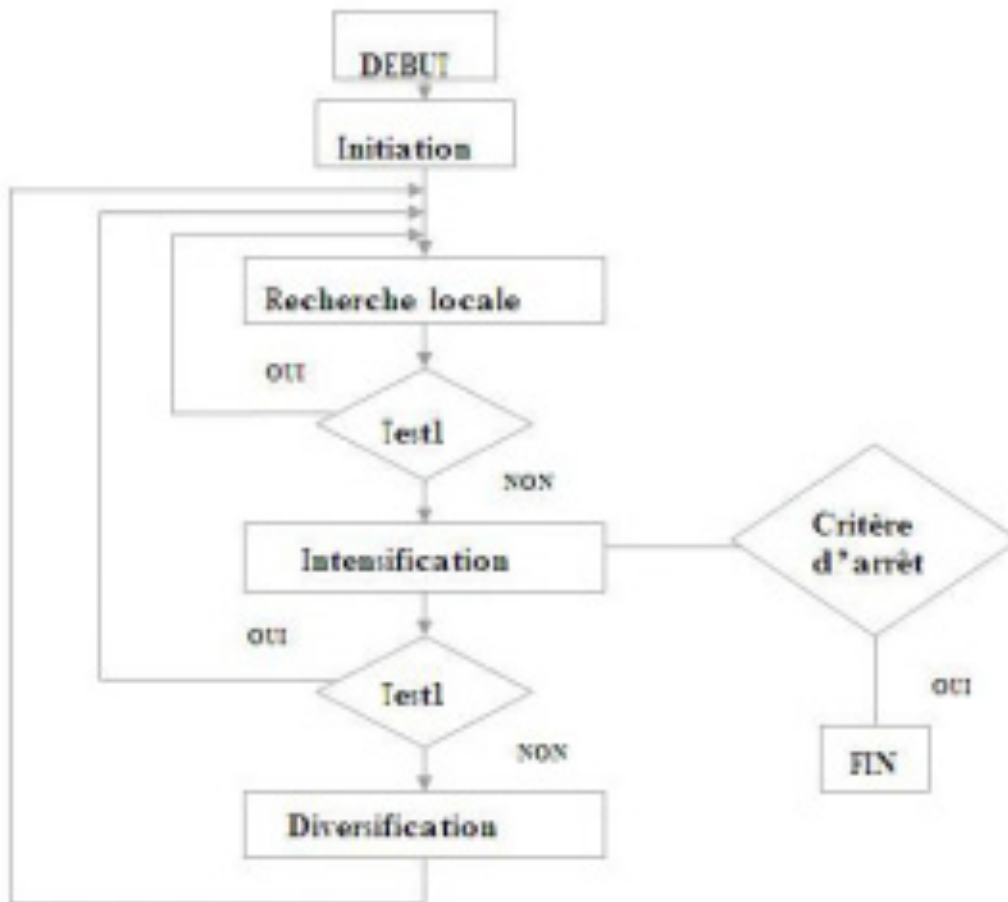


Figure 3 : Diagramme de transition de recherche tabou

### 4.3 Les algorithmes génétiques (AG)

Les algorithmes génétiques sont une abstraction de la théorie de l'évolution. Ils résolvent des problèmes n'ayant aucune méthode de résolution décrite précisément ou dont la solution exacte, si elle est connue, est trop complexe pour être calculée en un temps raisonnable. C'est notamment le cas quand des contraintes multiples, complexes doivent être satisfaites simultanément.

Le but des AG est de déterminer les extrêmes d'une fonction  $f : X \rightarrow \mathbb{R}$ , où  $X$  est un ensemble quelconque appelé espace de recherche et  $f$  est appelée fonction d'adaptation ou fonction d'évaluation. Cette fonction agit comme une «boite noire» pour l'AG.

Le premier pas dans l'implantation des algorithmes génétiques est de créer une population d'individus initiaux. En effet les algorithmes génétiques agissent sur une population d'individus, et non pas sur un individu isolé. Chaque individu de

la population est codé par un chromosome. Une population est donc un ensemble de chromosomes. Chaque chromosome code un point de l'espace de recherche. L'efficacité de l'algorithme génétique va donc dépendre du choix du codage d'un chromosome. A chaque étape (appelée génération), ces chromosomes se combinent, mutent, et sont sélectionnés en fonction de leur qualité à répondre au problème.

## **4.4 Le recuit simulé (RS)**

### **4.4.1 Principe générale**

La méthode du recuit simulé est inspirée du principe thermodynamique dans lequel les déplacements dans l'espace de recherche sont basés sur la distribution de Boltzmann. Le nom de cette méthode provient du domaine de la métallurgie où pour donner des formes au métal on le chauffe et on le refroidit, parfois de manière répétitive. Le recuit sert à obtenir des métaux plus malléables et plus résistants aux chocs.

Pour appliquer ce principe au problème de minimisation de coût, le processus de recherche peut être assimilé à un processus de recuit comme en métallurgie.

Le principe du recuit simulé est le suivant : à partir d'une première solution fixée par l'utilisateur, une nouvelle solution est générée selon une certaine fonction de distribution de probabilité  $g(x)$ , la valeur de la fonction à minimiser est calculée pour cette solution.

La solution est acceptée ou rejetée, selon la valeur d'une certaine fonction d'acceptation  $h(x)$ , qui est fonction d'un paramètre appelé «température», et de la différence des valeurs de la fonction à minimiser entre la solution courante et la solution précédente. La solution précédente devient la solution courante, la température est abaissée, et le processus recommence.

Initialement, la valeur de la température est élevée, et la nouvelle solution est acceptée près de la moitié du temps. Puis, au fur et à mesure que la température baisse, il devient de plus en plus improbable d'accepter une solution dont la valeur est plus grande que la valeur de la solution précédente, ce qui effectue bien une minimisation de la fonction. Le fait qu'une solution peut être acceptée parfois même si elle donne une valeur de la fonction plus grande que la valeur précédente permet à l'algorithme de sauter les minima locaux et de trouver le minimum global dans l'espace de définition de la fonction.

#### 4.4.2 Le paramètre température

La température est un paramètre de contrôle. Le but est d'avoir une température assez haute pour sauter les barrières et suffisamment basse pour être malgré tout attiré vers le minimum le plus profond. La loi selon laquelle la température décroît est également importante pour l'efficacité de l'algorithme, puisqu'elle doit aussi laisser le temps au système de tester le maximum de configurations pour être sûr d'obtenir le minimum global. Aussi, la température initiale doit-elle être suffisamment élevée pour que la descente en température soit aussi lente que possible.

Mais le problème de minima local n'est pas pour autant éliminé. Le recuit simulé est très dépendant de la structure de voisinage et le nombre de paramètres qui gère la recherche. Des techniques d'amélioration du RS ont été proposées, notamment la technique du « recuit très rapide ».

#### 4.5 Comparaison des différents algorithmes.

Nous présentons dans le tableau suivant une synthèse de la comparaison des trois algorithmes exposés dans ce chapitre :

<b>Recherche Tabou</b>	<b>Recuit simulé</b>	<b>Algorithme génétique</b>
<p><b>Avantages :</b> Absorption de plus grand nombre de problèmes d'optimisation. Résultat plus proche de l'optimal. Convergence rapide</p> <p><b>Inconvénients:</b> Le caractère probabiliste. Rapidité de convergence dépendant du choix de la population initiale.</p>	<p><b>Avantages:</b> Aboutit à un minimum global. Peut supporter plusieurs contraintes.</p> <p><b>Inconvénients:</b> Convergence assez lente. Sensible au réglage des paramètres. Temps de calculs relativement importants. La difficulté de définir une loi de descente de la température.</p>	<p><b>Avantages:</b> Aboutit à un minimum global. Moins sensible au changement des paramètres. Facile dans son implémentation. Peut être adapté à divers problèmes</p> <p><b>Inconvénients:</b> Convergence lente. Temps de recherche relativement lent.</p>



# **Chapitre 5 : Résolution de la problématique de localisation des concentrateurs par la méthode de « Recherche Tabou »**

## **5.1 Introduction**

Nous proposons dans ce chapitre de donner une approche de résolution de la problématique de localisation des concentrateurs formulée dans 4.1 en utilisant la méthode « Recherche Tabou » en supposant que les positions des MSC ne sont pas fixées, nous utilisons pour cela le langage java.

## **5.2 Plate forme informatique**

### **5.2.1 Choix de l'outil de programmation**

Le choix de Java est justifié par les raisons suivantes :

C'est un langage orienté objet ce qui simplifie la tâche de conception de l'application et qui facilite la compréhension rapide de l'application par d'autres personnes. Il donne la possibilité d'améliorer l'interface graphique.

### **5.2.2 Les modules utilisés dans l'application**

La tâche principale de l'application est de trouver une architecture globale du réseau qui réalisera un coût minimal. Cette architecture est construite en passant par deux étapes :

- La première étape spécifie l'architecture entre les BTS et les BSC et comporte la détermination du nombre de BSC à déployer dans le réseau, leurs positions et les connexions qu'ils établissent avec les BTS.

- La deuxième spécifie une architecture entre les BSC et les MSC en précisant le nombre de MSC devant être utilisés, leurs positions et les connexions qu'ils établissent avec les BSC.

Dans les deux cas nous sommes appelés à résoudre un problème combinatoire entre des sites terminaux et des sites concentrateurs. La formulation du problème est déjà donnée dans le chapitre précédent. Il s'agit de minimiser une fonction coût composée des coûts de raccordement et des coûts d'implantation des concentrateurs sous la contrainte de capacité de raccordement.

Une solution quelconque du problème, c'est-à-dire une architecture entre sites terminaux et sites concentrateurs, est une instance de la classe Solution déclarée en java

comme suit :

```
class Solution
{
    boolean[] y;
    boolean[][] x;
    Solution(int n,int p)
    {
        y=new boolean[p];
        x=new boolean[n][p];
    }
}
```

Y est un tableau de booléens (true ou false) qui indique que si  $y[j]=true$  alors un concentrateur est installé dans le site j et  $y[j]=false$  sinon.

X est un tableau de deux dimensions indiquant l'état des liaisons entre les sites terminaux et les concentrateurs : On a donc  $x[i][j]=true$  s'il existe une liaison entre le site terminal numéro i et le site concentrateur numéro j.

On parlera dans la suite des fonctions utilisées pour traiter la première problématique qu'est le positionnement des BSC et l'affectation des BTS au BSC, les fonctions utilisées pour la deuxième problématique sont presque les mêmes.

Les fonctions utilisées sont les suivantes :

- int getNombreBSC() : Cette fonction retourne un entier indiquant le nombre de BSC qu'il faut avoir pour satisfaire 60 pourcent des demandes en trafic provenant des BTS.
- Solution solution\_initiale() : cette fonction génère une solution initiale en affectant les BTS aux BSC les plus proche sans dépasser les capacités de ces derniers.
- Solution mouvement\_connexion(Solution s,int indice\_BTS,int indice BSC) génère une solution voisine de la solution s donnée en paramètre en connectant la BTS au BSC indiqués par leurs indices.
- Solution mouvement\_position\_BSC(Solution s,int j,int k) : génère une solution voisine de s en commutant les valeurs de  $y[j]$  et  $y[k]$  de s, en d'autres termes, permet de choisir de nouvelles positions des BSC parmi les sites possibles.
- Double cout(Solution s) : Calcule le coût d'une solution composé du

coût de liaisons et le coût d'implantation des BSC.

- Solution meilleur\_voisin\_connexion(Solution s,int nombre\_voisin) : C'est une fonction qui retourne la meilleure des solutions voisines de s obtenues par la fonction mouvement\_connexion, le paramètre nombre\_voisin indique le nombre de voisin qu'on génère à partir de s et qu'on compare pour obtenir la meilleure.
- Solution meilleur\_voisin\_BSC(Solution s,int nombre\_voisin) : Retourne la meilleure solution voisines de s obtenues par la fonction mouvement\_position\_BSC().
- Solution RechercheTaboue(Solution s\_initiale,int nombre\_voisin\_conexion,int nombre\_voisin\_position\_BSC,int taille\_listeTaboue,int Kmax)

Cette fonction est celle permettant d'obtenir la solution proposée par la RT, le premier paramètre est une solution initiale obtenue par la fonction : solution\_initiale(), le deuxième et le troisième représentent le nombre de voisins à générer pour chaque itération de l'algorithme. Le quatrième indique la taille de la liste tabou qu'il ne faut pas dépasser sinon la recherche s'arrête. Et le dernier indique le seuil maximal de la différence entre deux entiers nbiter et bestiter : nbiter s'incrémente à chaque itération et bestiter devient égal à nbiter si on trouve une solution plus optimale que la dernière trouvée.

Voici l'algorithme général de recherche tabou :

Initialisation : Choisir une solution initiale S dans X (l'ensemble de solutions)  $S^* := S$  ( $S^*$  est la meilleure solution obtenue jusqu'ici)

Nbiter :=0 (nbiter est le compteur des itérations)

Bestiter :=0 (bestiter est le numéro de l'itération à laquelle on a obtenu la dernière amélioration, i.e le dernier  $S^*$ )

T := cp (T est la liste Taboue)

Continuer :=vrai

Tant que Continuer :=vrai

Si (nbiter-bestiter>Kmax) ou (V-T=cp) ou taille(T)>Taille\_MAX

Alors Continuer :=faux

Sinon : nbiter=nbiter+1

Générer V inclus dans N(s)

Trouver la meilleure solution  $S'$  dans V tel que :

Cout( $S'$ )=min(cout(S)) avec  $S \in V-T$

```

    S := S'
    Mette à jour T.
    Si cout(S') < cout(S) alors S* := S
    Bestiter :=nbiter
    Fin si
  Fin si
Fin tant que

```

### 5.2.3 La solution proposée par l'application

Dans ce paragraphe nous allons présenter les différentes étapes par lesquelles on générera une solution acceptable (et qui peut être l'optimale) proposée par l'algorithme de recherche tabou.

L'utilisateur doit saisir tout d'abord les paramètres des BTS.

Ensuite, il doit saisir les paramètres des sites possibles des BSC. Parmi ces paramètres on trouve la capacité en TRX des BSC et la capacité de raccordement. Ces capacités peuvent être changées par l'utilisateur.

Après la phase de saisie des paramètres, on peut lancer l'algorithme.

Le coût de cette solution, qui est composé du coût d'implantation des BSC plus le coût des liaisons entre BTS et BSC, est mentionné plus bas à droite de l'application.

Nous procédons ensuite à la recherche d'une architecture de liens entre BSC et MSC qui réalise un coût minimal puisque nous avons déjà trouvé les positions des BSC à déployer dans le réseau. Pour cela l'utilisateur doit saisir les paramètres concernant les sites possibles des MSC. Ces paramètres sont les positions en longitude et latitudes des sites possibles, le coût d'implantation d'un MSC dans chaque site et la capacité de raccordement des MSC.

## 5.3 Conclusion

Dans ce chapitre nous exposons notre travail d'optimisation qui consiste à appliquer la méthode de recherche Tabou au problème de positionnement des concentrateurs en utilisant le langage Java, l'utilisateur de l'application peut enregistrer les paramètres des BTS, des sites possibles des BSC, des sites possibles des MSC et ceux de l'algorithme qui peuvent être utilisés ultérieurement.

Pour plus d'information, vous pouvez voir en annexe le code source des modules utilisés accompagné de quelques explications.

## Conclusion générale

La planification du NSS est une tâche complexe pouvant être divisée en plusieurs sous tâches. La modélisation mathématique de ces problématiques les place parmi les problèmes NP-complexes, ce qui oblige à passer par des méthodes de résolution non exactes.

Nous donnons la formulation de ces problématiques, et en particulier nous étudions le problème de localisation des BSC et MSC en proposant comme solution la méthode heuristique dite de recherche Tabou.

Ce projet de fin d'étude nous permet d'étudier à la fois les algorithmes d'optimisation (la recherche Tabou, le recuit simulé, les algorithmes génétiques) qui font l'objet des études et des travaux à l'échelle internationale dans le domaine des télécommunications, il nous permet aussi de se lancer dans la programmation Java en modélisant le problème de positionnement des concentrateurs et la mise en œuvre des connexions avec les sites terminaux, qu'il s'agit du problème entre les BTS et BSC ou celui entre les BSC et MSC, et cela en utilisant la méthode de recherche Tabou.

Comme perspective de notre travail, nous proposons l'étude du problème avec les deux autres méthodes à savoir le recuit simulé et les algorithmes génétiques pour faire une comparaison réelle des performances de chacune d'elles concernant le problème combinatoire traité.

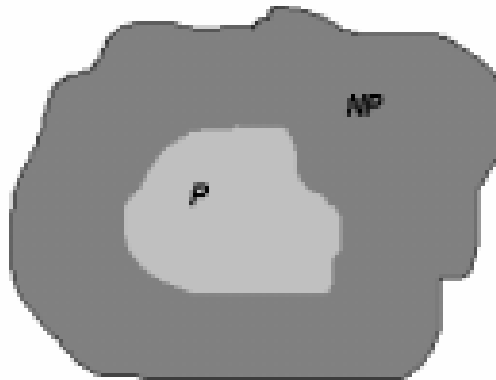
# ANNEXES

## ANNEXE A : Problèmes NP-Difficiles

Nous classifions le problème de positionnement des BSC et MSC parmi les problèmes NP difficiles. Nous présentons ici une description de cette classe de problèmes.

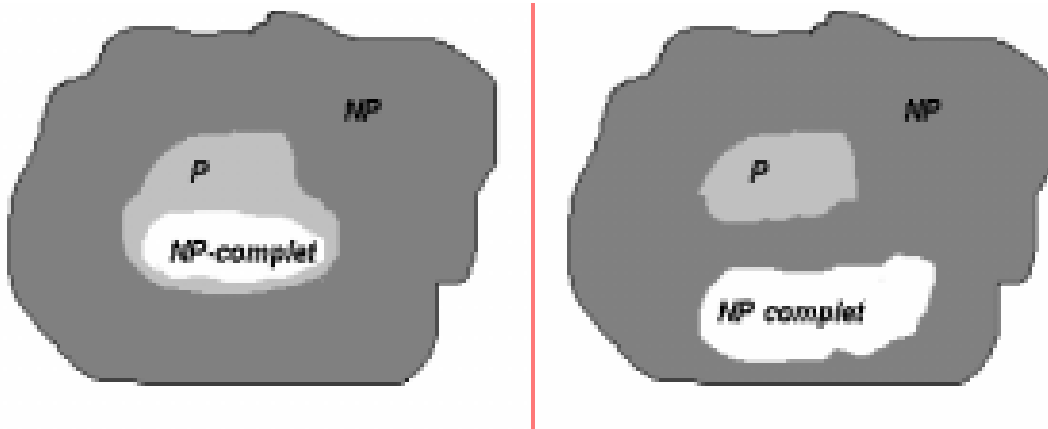
Une des tâches importantes d'un informaticien théoricien est de classifier les problèmes à résoudre dans une des classes suivantes :

- Classe des problèmes pour lesquels il existe un algorithme de résolution polynomial, dénoté P.
- Classe des problèmes qui n'ont pas d'algorithme de résolution polynomial, mais pour lesquels on est capable de vérifier dans un temps polynomial si s (une proposition de solution) est ou non une solution, cette classe est dénotée par NP.



Dans le monde informatique il existe un groupe de problèmes pour lesquels on peut vérifier dans un temps polynomial si s (une proposition de solution) est ou non une solution, mais il n'existe pas d'algorithmes polynomiaux de résolution et en même temps on n'a pas été capable de démontrer que de tels algorithmes n'existent pas.

Ils forment la classe des problèmes NP-complet. La question qui n'est pas encore décidée est la suivante :



**Exemple :** Décider si on veut trouver pour le graphe  $G=(V,E)$ , le cycle qui passe par tous les nœuds du graphe une seule et unique fois (exception pour le premier = dernier nœud du cycle) qui a la longueur minimale (chaque arrête ou arc du graphe ayant un coût).

C'est le problème du commis voyageur.

Ce problème est un problème d'optimisation. On doit chercher parmi les coûts associés aux cycles passant par tous les nœuds du graphe, le coût minimal et identifier le cycle associé. C'est un problème NP-difficile.

On peut associer au problème du commis voyageur un problème de décision qui est un peu moins compliqué.

Soit un graphe  $G$ . Peut on trouver un cycle qui passe par tous ses nœuds et qui a un coût plus petit que  $K$  ?

Si quelqu'un propose une solution  $s$ , on peut vérifier dans un temps polynomial si la valeur de  $s$  est ou non inférieur à  $K$ .

Pour trouver si on peut répondre ou non à la question du problème, on peut considérer les cycles possibles en  $G$ , calculer leurs coûts et comparer chacun avec  $K$ . Dès qu'on trouve un qui a le coût inférieur à  $K$ , on a trouvé la solution du problème. Si chaque coût calculé est plus grand que  $K$ , alors pour répondre non à la question du problème on doit vérifier tous les cycles du graphe. Ce problème est NP complet.

## ANNEXE B : Code source des modules les plus importants

Dans cette annexe, nous ne présentons que le code des modules principaux du problème de localisation des BSC et MSC puisque les fonctions et classes de l'affichage sont trop longues.

Voici tout d'abord les variables globales utilisées :

```
/** Les listes des BTS, des sites des BSC possibles, des BSC, des sites des MSC possibles et des MSC */
```

```
ArrayList listeBTS = new ArrayList();
```

```
ArrayList listeBSC_possible = new ArrayList();
```

```
ArrayList listeBSC = new ArrayList();
```

```
ArrayList listeMSC_possible=new ArrayList(); ArrayList listeMSC = new ArrayList();
```

```
/** La liste des solutions taboues */
```

```
ArrayList listeTabou=new ArrayList();
```

```
/** Les capacités des BSC et des MSC */
```

```
int capacite_BSC_TRX;
```

```
int capacite_BSC_connexion;
```

```
int capacite_MSC_connexion;
```

```
/** Ces deux entiers enregistrent les nombres des BSC et MSC choisis parmi les BSC et MSC possibles */
```

```
int nombre_BSC;
```

```
int nombre_MSC;
```

```
/** Variables de type Solution qui enregistrent les solutions initiale, courante et optimale obtenues pour les BSC et les MSC */
```

```
Solution s_initiale_BSC; Solution s_courante_BSC; Solution s_optimale_BSC;
```

```
Solution s_initiale_MSC; Solution s_courante_MSC; Solution s_optimale_MSC;
```

```
/** Paramètres de l'algorithme */
```

```
int taille_tabou;
```

```
int voisin_connexion;
```

```
int voisin_position_concentrateur;
```

```
int MaxPlusOptimale;
```

```
int NombreDiversification;
```

```
int nombre_tout_voisin_tabou;
```

Et voici les classes utilisées :

```
/** Classe des Solutions */
```

```
/** Cette classe contient deux tableaux boolean (true ou false), le premier y[] est unidimensionnel et permet de signaler les sites où on a installé un BSC parmi les sites possibles. Et le deuxième x[][] est bidimensionnel et permet de signaler si une connexion entre un BTS et BSC existe ou non */
```



```

Class Solution
{
boolean[] y; boolean[][] x; Solution(int n,int p)
{
y=new boolean[p];
x=new boolean[n][p];
}
}
/*****Classe des BTS *****/
class BTS {
int nombreTRX;
int left,top,width,height ;
}
class BSC_possible {
int cout_implantation;
int left,top,width,height ;
}
/*****Classe des BSC possibles *****/
class BSC_possible
{
double cout_implantation;
int left,top,width,height ;
}
/*****Classe des MSC possibles *****/
class MSC_possible
{
double cout_implantation;
int left,top,width,height ;
}
/***** Fonction qui calcule le coût d'implantation *****/
double cout_implantation(Solution s)
{
int nb_bsc_possible; nb_bsc_possible=canvas.listeBSC_possible.size(); double
somme_cout_implantation=0;
for(int i=0;i<nb_bsc_possible;i++)
{
if(s.y[i]==true)
{
somme_cout_implantation=somme_cout_implantation+((BSC_possible)canva
s.listeBSC_possible.get(i)).cout_implantation;
}
}
return somme_cout_implantation;
}

```

Et le coût total s'obtient en faisant la somme des coûts de liaison et d'implantation :

```
/****** Fonction calculant le coût total d'une solution *****/
```

```
double cout(Solution s)
{
return cout_implantation(s)+cout_liaison(s);
}
```

Une fois nous avons pu générer une solution voisine d'une solution par l'une des deux critères de voisinages, on peut programmer une fonction qui nous donne le meilleur voisin pour chaque type de voisinage.

```
/* fonction qui retourne la meilleur solution voisine par connexion d'une solution s */
```

```
Solution meilleur_voisin_connexion(Solution s,int nombre_voisin)
{
int nb_bts; nb_bts=canvas.listeBTS.size(); int nb_bsc_possible;
nb_bsc_possible=canvas.listeBSC_possible.size();
Solution meilleur_voisin=new Solution(nb_bts,nb_bsc_possible);
if(nb_bts>0&&nb_bsc_possible>0)
{
int nb_bsc;
nb_bsc=canvas.nombre_BSC;
int concentrateur[] = new int[nb_bsc];
int k;
{
Solution solution=new Solution(n,p)
for(int j=0;j<p;j++)
{
for(int i=0;i<n;i++)
{
solution.x[i][j]=s.x[i][j];
}
solution.y[j]=s.y[j];
}
for(int i=0;i<p;i++)
{
if(solution.x[indice_terminal][i]==true)
{
solution.x[indice_terminal][i]=false;
}
}
solution.x[indice_terminal][indice_concentrateur]=true;
return solution;
}
```

```
/****** Fonction permettant de donner une solution voisine de s en spécifiant deux indice  
des sites des BSC possibles pour faire la transposition de positionnement des BSC entre ces  
deux sites *****/
```

```
Solution mouvement_position_concentrateur(Solution s,int u,int v,int n,int p)
```

```
{  
boolean b;
```

```
Solution solution = new Solution(n,p);
```

```
for(int j=0;j<p;j++)
```

```
{
```

```
for(int i=0;i<n;i++)
```

```
{
```

```
solution.x[i][j]=s.x[i][j];
```

```
}
```

```
solution.y[j]=s.y[j];
```

```
}
```

```
b=solution.y[u]; solution.y[u]=solution.y[v]; solution.y[v]=b;
```

```
for(int i=0;i<n;i++)
```

```
{
```

```
if(s.x[i][u]==true)
```

```
{
```

```
solution.x[i][u]=false;
```

```
solution.x[i][v]=true;
```

```
}
```

```
if(s.x[i][v]==true)
```

```
{
```

```
solution.x[i][v]=false;
```

```
solution.x[i][u]=true;
```

```
}
```

```
}
```

```
return solution;
```

```
}
```

```
int indice_terminal[] = new int[nombre_voisin];
```

```
int indice_concentrateur[] = new int[nombre_voisin];
```

```
k=0;
```

```
for(int i=0;i<nb_bsc_possible;i++)
```

```
{
```

```
if(s.y[i]==true)
```

```
{
```

```
concentrateur[k]=i;
```

```
k++;
```

```
}
```

```

}
Random r=new Random();
for(int i=0;i<nombre_voisin;i++)
{
indice_terminal[i] = r.nextInt(nb_bts);
indice_concentrateur[i] = concentrateur[r.nextInt(nb_bsc)];
}
Solution[] voisinage = new Solution[nombre_voisin];
meilleur_voisin=mouvement_connexion(s,indice_terminal[0],indice_concentra
teur[0],nb_bts,nb_bsc_possible);
for(int i=1;i<nombre_voisin;i++)
{
voisinage[i]=mouvement_connexion(s,indice_terminal[i],indice_concentrateur[i],nb_bts,n
b_bsc_possible);
if((cout_liaison(voisinage[i])<cout_liaison(meilleur_voisin))&&conforme_capaci
te(voisinage[i]))
{
meilleur_voisin=voisinage[i];
}
}
}
return meilleur_voisin;
}
/***** Fonction qui retourne la meilleur solution voisine par position de
concentrateurs d'une solution s *****/
Solution meilleur_voisin_position_concentrateur(Solution s,int nombre_voisin)
{
int nb_bts; nb_bts=canvas.listeBTS.size(); int nb_bsc_possible;
nb_bsc_possible=canvas.listeBSC_possible.size();
Solution meilleur_voisin=new Solution(nb_bts,nb_bsc_possible);
if(nb_bts>0&&nb_bsc_possible>0)
{
int nb_bsc;
nb_bsc=canvas.nombre_BSC;
int concentrateur[] = new int[nb_bsc];
int k;
k=0;
for(int i=0;i<nb_bsc_possible;i++)
{
if(s.y[i]==true)
{
concentrateur[k]=i;
k++;
}
}
}

```

```

}
int pos1[] = new int[nombre_voisin];

int pos2[] = new int[nombre_voisin];
Random r=new Random();
for(int i=0;i<nombre_voisin;i++)
{
pos1[i] = r.nextInt(nb_bsc_possible);
pos2[i] = concentrateur[r.nextInt(nb_bsc)];
}
Solution[] voisinage = new Solution[nombre_voisin];
meilleur_voisin=mouvement_position_concentrateur(s,pos1[0],pos2[0],nb_bts,
nb_bsc_possible);
for(int i=1;i<nombre_voisin;i++)
{
voisinage[i]=mouvement_position_concentrateur(s,pos1[i],pos2[i],nb_bts,nb_bsc_possible);
if((cout(voisinage[i])<cout(meilleur_voisin))&&conforme_capacite(voisinage[i]))
{
meilleur_voisin=voisinage[i];
}
}
}
return meilleur_voisin;
}

```

## ANNEXE C : Abréviations

<b>AG</b>	Algorithmes Génétiques
<b>AuC</b>	Authentication Center
<b>BSC</b>	Base Station Controller
<b>BSS</b>	Base Station Sub-System
<b>BTS</b>	Base Tranceiver Station
<b>EIR</b>	Equipement Identity Register
<b>GSM</b>	Global System for Mobile Communications (Groupe Spécial Mobile)
<b>GMSC</b>	Gateway Mobile Switching Center
<b>HLR</b>	Home Location Register
<b>IMEI</b>	Intenational Mobile Equipement Identity
<b>IMSI</b>	International Mobile Subscriber Identity
<b>LA</b>	Local Area
<b>MS</b>	Mobile Station
<b>MSC</b>	Mobile Services Switching Center ou Mobile Switchnig Center
<b>MSISDN</b>	Mobile Subscriber International ISDN Number
<b>MSRN</b>	Mobile Station Roaming Number
<b>NSS</b>	Network Sub-System
<b>OMC</b>	Operation and Maintenance Center
<b>RS</b>	Recuit Simulé
<b>RTCP</b>	Réseau Téléphonique Commuté Public
<b>SIM</b>	Subscriber Identity Mobile
<b>TDMA</b>	Temporal-Division Multiple Access
<b>VLR</b>	Visitor Location Register

## Bibliographie

- Sami Tabbane, « Ingénierie des réseaux cellulaires », édition: HERMES, 2002
- « Planification du sous-système réseau » transparent du cours cycle d'ingénieurs d'état troisième année à l'INPT.
- « Affectation heuristique de cellules à des commutateurs dans les réseaux cellulaires pour mobiles » ANN. TELECOMMUN. , 56, no 3-4, 2001 par F.HOUETO et S.PIERRE.
- Xavier Lagrange, Philippe Godlewski, Sami Tabbane « Réseaux GSM-DCS », troisième édition, HERMES, 1997.
- Mémoire de fin d'études : Planification du réseau fixe du GSM, Lamallam Abdessamad, Institut National des Postes et Télécommunications (Royaume du Maroc), 2004.
- <http://www.technologuepro.com/cours-informatique/cat-1-cours-reseaux-telecommunication>