



REPUBLIQUE ALGERIENNE  
DEMOCRATIQUE ET POPULAIRE  
MINISTRE DE L'ENSEIGNEMENT  
SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE



UNIVERSITE CHIKH LARBI TEBESSA - TEBESSA  
FACULTE DES SCIENCES ET TECHNOLOGIES  
DEPARTEMENT DE GENIE ELECTRIQUE

## MEMOIRE

DE FIN D'ETUDES POUR L'OBTENTION DU DIPLOME DE MASTER EN  
TELECOMMUNICATIONS

### THEME

# CODAGE REED SOLOMON POUR LES INTERFEROGRAMMES inSAR

Présenté par le binôme :

- DERBAL Nabila
- ALI Hakima

Devant le jury :

- |                      |           |
|----------------------|-----------|
| - MERAOUMIA Abdallah | Président |
| - BENTAHAR Tarek     | Encadreur |
| - GATTAL Azzedine    | Examineur |

# *DEDICACE*

Je dédie ce modeste travail à :

Ma très chère mère

Quoi que je fasse ou que je dise, je ne saurai point te remercier comme il se doit, ton affection me couvre, ta bienveillance me guide et ta présence à mes coté a toujours été ma source de force pour affronter les différents obstacle .. je t'aime maman

Mon très cher père

Mon support dans la vie , tu as toujours été à mes coté pour me soutenir et m'encourager.. je t'aime papi

A mes frères et sœurs, Zineb et Abde-el madjid

A la mémoire de mon chère frère Allah yerhmou...

Et spécialement a Salah eddine

A mes amies (Maissoun, Amira ,Yassmine,Hakima,Sofia)

et toute personne qui ma aidée de loin ou de prés.

*Derbal Nabila*

# *DEDICACE*

Je dédie ce travail

A ma maman qui m'a soutenu et encouragé durant ces années d'études.

Qu'elle trouve ici le témoignage de ma profonde reconnaissance.

A ma sœur, mes frères.

A tous mes amis qui m'ont toujours encouragé, et à qui je souhaite plus  
de succès.

A tous ceux que j'aime.

*Ali Hakima*

# Remerciement

Avant tout nous remercions dieu pour tout et pour la volonté pour reprendre nos études.

nous tenons à remercier nos parents d'avoir sacrifiés leur vie pour notre bien.

notre profonde gratitude s'adresse tout particulièrement à notre rapporteur de mémoire, monsieur Bentahar Tarek pour avoir accepté de suivre la réalisation de notre mémoire, aussi pour ses conseils et son aide qu'il nous apporté tout au long de ce travail.

nos remerciements vont à tous les membres de jury qui ont accepté de débattre notre sujet de mémoire.

en fin, nous adressons nos vifs remerciements aux enseignants de notre spécialité.

D. Nabila et A. Hakima

# Table des Matières

Dédicaces.....	i
Remerciements.....	iii
Table de matières .....	iv
Liste des figures .....	viii
Liste des tableaux.....	x
Introduction générale.....	1

## Chapitre I : Généralité sur le système d'imagerie inSAR

I.1 Introduction.....	3
I.2 C'est quoi le système inSAR .....	3
I.2.1 L'antenne radar.....	4
I.2.2 Bandes de fréquences .....	4
I.2.3 Quelques satellites SAR et inSAR .....	5
I.3 Principe de SAR.....	6
I.3.1 La Résolution.....	6
I.3.2 les défauts de projection .....	9
I.3.3 le signal et les images SAR.....	10
I.4 Principe de l'inSAR.....	11
I.5 Signal de corrélation et les images inSAR.....	14
I.5.1 Décorrélacion spatiale .....	15
I.5.2 Décorrélacion temporelle .....	16
I.5.3 les images inSAR .....	16
I.6 Conclusion .....	19

## Chapitre II : Étude, analyse et comparaison de différentes méthodes de déroulement de phase

II.1 Introduction .....	20
-------------------------	----

<b>II.2</b> C'est quoi le déroulement de phase ?.....	<b>21</b>
<b>II.2.1</b> formulations.....	<b>21</b>
<b>II.2.2</b> La condition d'Itoh .....	<b>22</b>
<b>II.3</b> Déroulement par ajustement.....	<b>23</b>
<b>II.3.1</b> traitement 1D.....	<b>24</b>
<b>II.3.2</b> traitement 2D.....	<b>25</b>
<b>II.4</b> autres approches de déroulement .....	<b>28</b>
<b>II.4.1</b> Les méthodes de Branch-cut .....	<b>28</b>
<b>II.4.1.1</b> La méthode de Goldstein .....	<b>28</b>
<b>II.4.1.2</b> Méthode <i>Branch-Cut</i> guidée par la fonction pseudo corrélation .....	<b>33</b>
<b>II.4.1.3</b> Méthode Branch-Cut d'appariement par distance.....	<b>35</b>
<b>II.4.2</b> Les méthodes de qualité guidée et la croissance de la région .....	<b>37</b>
<b>II.4.2.1</b> La méthode de Flynn .....	<b>37</b>
<b>II.4.2.2</b> La méthode de Wei Xu .....	<b>41</b>
<b>II.4.2.3</b> La méthode de quantification de la qualité.....	<b>44</b>
<b>II.4.3</b> Les méthodes de norme de minimisation.....	<b>45</b>
<b>II.4.3.1</b> les normes $L^p$ .....	<b>45</b>
<b>II.4.3.2</b> la norme $L^2$ .....	<b>46</b>
<b>II.4.3.3</b> La norme $L^1$ .....	<b>47</b>
<b>II.4.3.4</b> les normes de petite valeur de p.....	<b>47</b>
<b>II.5</b> Conclusion.....	<b>47</b>
 <b>Chapitre III : Codage de canal</b>	
<b>III.1</b> Introduction.....	<b>48</b>

<b>III.2</b>	Codage de canal .....	<b>48</b>
<b>III.2.1</b>	Technique de codage de canal .....	<b>48</b>
<b>III.2.2</b>	Les applications de codage de canal.....	<b>49</b>
<b>III.3</b>	Classification des codes Correcteurs .....	<b>50</b>
<b>III.4</b>	Les codes linéaires.....	<b>51</b>
<b>III.4.1</b>	Matrice génératrice.....	<b>51</b>
<b>III.4.2</b>	Matrice de contrôle d'un code linéaire .....	<b>51</b>
<b>III.4.3</b>	Distance minimale d'un code linéaire.....	<b>52</b>
<b>III.4.4</b>	Décodage d'un code linéaire .....	<b>53</b>
<b>III.5</b>	Les codes cycliques .....	<b>53</b>
<b>III.5.1</b>	Rappels sur les polynômes.....	<b>53</b>
<b>III.5.2</b>	Définitions d'un code cyclique .....	<b>54</b>
<b>III.6</b>	Problème principal du codage.....	<b>55</b>
<b>III.7</b>	les Corps de Galois GF (Généralités) .....	<b>55</b>
<b>III.7.1</b>	Propriétés d'un corps fini.....	<b>55</b>
<b>III.7.2</b>	Construction d'un corps fini .....	<b>55</b>
<b>III.8</b>	Codage des codes de Reed Solomon.....	<b>56</b>
<b>III.8.1</b>	Définition.....	<b>56</b>
<b>III.8.2</b>	Avantages des codes de Reed Solomon... ..	<b>57</b>
<b>III.8.2</b>	Avantages des codes de Reed Solomon.....	<b>57</b>
<b>III.8.3</b>	Principes de codage Reed Solomon .....	<b>57</b>
<b>III.8.3.1</b>	Polynôme générateur.....	<b>57</b>
<b>III.8.3.2</b>	Schéma général du codage Reed Solomon... ..	<b>59</b>
<b>III.8.4</b>	Décodage des codes de Reed Solomon .....	<b>59</b>

III.8.4.1	Calcul du syndrome.....	60
III.8.4.2	Évaluation du polynôme de locations d'erreurs .....	61
III.8.4.3	Equation fondamentale.....	61
III.8.4.4	Algorithme D'Euclide .....	62
III.8.4.5	Algorithme de Reed Solomon .....	63
III.8.4.6	Schéma d'Horner .....	63
III.8.4.7	Détection d'erreurs dont le nombre est supérieur à $t$ .....	65
III.9	Applications des codes Reed-Solomon .....	65
III.10	Conclusion .....	66
<b>Chapitre IV : Analyse de code Reed Solomon</b>		
IV1	Introduction.....	67
IV2	Rappel sur le code Reed- Solomon.....	69
IV.2.1	Description.....	69
IV.2.2	Analyse du taux de code RS.....	71
IV3	Conclusion.....	74
<b>Conclusion général.....</b>		<b>75</b>
<b>Bibliographies .....</b>		<b>77</b>
<b>Acronymes.....</b>		<b>83</b>
<b>Résumé.....</b>		<b>85</b>

# Liste des Figures

Figures	Page
<b>I.1</b> Différence entre SAR et inSAR.....	5
<b>I.2</b> L'ouverture rayonnante rectangulaire.....	5
<b>I.3</b> Notion d'illumination en azimute et en distance.....	7
<b>I.4</b> La durée d'impulsion et la résolution distale.....	8
<b>I.5</b> Les défauts de projection parabolique de l'onde radar.....	9
<b>I.6</b> Images SAR .....	11
<b>I.7</b> L'altitude de point imagé et la différence de trajet.....	12
<b>I.8</b> La différence d'altitude et la différence de la distance radiale.....	13
<b>I.9</b> Images inSAR.....	14
<b>I.10</b> Les angles d'incidence et la décorrélation spatiale.....	15
<b>I.11</b> Exemple d'image inSAR de Vatnajökull ERS-1/2 (tandem data) ESA.....	19
<b>II.1</b> Problème non-unicité de déroulement selon le parcours suivi et le point de départ.....	21
<b>II.2</b> Traitement unidimensionnel pour une section de la surface, condition d'Itoh satisfaite.....	24
<b>II.3</b> Traitement unidimensionnel pour une section de la surface, violation de la condition d'Itoh.....	25
<b>II.4</b> Les deux modes simples de la méthode d'ajustement.....	26
<b>II.5</b> Les deux images générées pour des sauts de phase situés entre $x_{50} \leq x_i \leq x_{60}$ $y_{40} \leq y_j \leq y_{45}$ .....	27
<b>II.6</b> Les deux images générées pour des sauts de phase situés entre $x_{35} \leq x_i \leq x_{40}$ $y_{90} \leq y_j \leq y_{100}$ .....	27
<b>II.7</b> Le déroulement avec résidus et les lignes distordues.	28
<b>II.8</b> Test d'algorithme de déroulement Goldstein pour un interférogramme pics à franges fermées. ....	30
<b>II.9</b> Test d'algorithme de déroulement Goldstein pour un interférogramme ondes à franges ouvertes. ....	31

<b>II.10</b>	Test d'algorithme de déroulement Goldstein pour une partie d'un interférogramme réel d'ERS-1. ....	32
<b>II.11</b>	l'image de phase originale avec discontinuités raides, l'interférogramme correspondant.....	34
<b>II.12</b>	la mappe de pseudo-corrélation, la mappe de guide obtenue par la détection des bords. ....	34
<b>II.13</b>	la mappe des branches guidées par la pseudo-corrélation, la mappes des branches de Goldstein, l'image de phase déroulée après le placement de branches par GPC, l'image de phase déroulée après le placement de branches de Goldstein.....	35
<b>II.14</b>	Comparaison entre la mappe des branches de Goldstein et AD .....	36
<b>II.15</b>	Test d'algorithme de déroulement de Flynn pour un interférogramme pics à franges fermées .....	39
<b>II.16</b>	Test d'algorithme de déroulement de Flynn pour un interférogramme ondes à franges ouvertes. ....	40
<b>II.17</b>	Test d'algorithme de déroulement de Flynn pour un interférogramme ondes à franges ouvertes .....	41
<b>II.18</b>	Les prédictions des directions pour le pixel à croître et ses voisins.....	42
<b>III.1</b>	La hiérarchie des codes correcteurs .....	51
<b>III.2</b>	Mot-code d'un code de Reed Salomon $RS(n,k)$ .....	57
<b>III.3</b>	Schéma de codage .....	59
<b>IV.1</b>	Les étapes générales de décodage.....	70

# Liste des tableaux

	<i>Page</i>
<b>I.1</b> Les bandes de fréquences SAR et inSAR.....	6
<b>I.2</b> Quelques satellites SAR.....	7
<b>IV.1</b> le code rate par rapport au code spécification	72
<b>IV.2</b> le MSE avec longueurs du mot-code n=15.....	73
<b>IV.3</b> le MSE avec longueurs du mot-code n=63.....	73

# *Introduction Générale*

# Introduction générale

Depuis plusieurs décennies l'homme a porté un grand intérêt à l'imagerie de surface de la terre dans le but de collecter les informations topographiques telle que l'allure des reliefs ou de surveiller tout changement due à un phénomène de source humaine ou naturelle tels que l'impact de mouvement séismique et tectonique, les éruptions volcaniques, l'effet de changement climatique (température) et le gisement [1-3]. Ces différentes applications ont vu un substantiel avancement à l'avènement des systèmes d'imagerie radar en générale et inSAR (*Interferometric Synthetic Aperture Radar*) en particulier [3-12] car ils peuvent fonctionner le jour et la nuit et dans toutes les conditions météorologiques grâce à la possession de leurs propres sources d'illumination qui est l'onde Électromagnétique (EM) contrairement aux systèmes d'imagerie optique qui dépendent totalement au rayonnement solaire et à la clarté du milieu de propagation dans lequel la présence des nuages, brouillard, précipitation ou tout autres particules nuisent à l'acquisition correctes des images.

Le système inSAR est le seul système capable d'exploiter l'information de phase incluse dans le signal et de générer les modèles numériques du terrain (*Digital Elevation Model DEM*) interprétant les altitudes des points imagés en proportionnalité avec la phase interférométriques, mais cette phase mesurée est modulo  $2\pi$  ou autrement dit elle est roulée ; ce qui rend l'information d'altitude limitée par une altitude d'ambiguïté permettant de former des franges dans les images de phase interférométriques (nommées interférogrammes), alors pour avoir les altitudes réelles des points imagés nous devons passer de la phase modulo  $2\pi$  (Roulée/*Wrapped*) à la phase réelle (déroulée/*Unwrapped*) par un processus appelé déroulement de phase [13- 15]. La réussite de déroulement de phase dépend de bruits dans l'interférogramme communément appelés les résidus. Si l'interférogramme est sans résidus ; ce qui est carrément impossible en réalité ; le déroulement n'est qu'une simple intégration des gradients roulés. Mais avec les résidus le déroulement devient une tâche difficile voire même impossible pour un certain seuil à cause de la propagation d'erreurs lors de l'opération de déroulement en formant des lignes distordues le long des parcours d'intégration choisis dans l'image. Donc une approche fiable de traitement pour limiter ou éliminer l'effet propagatif

des résidus nous oblige de perfectionner le filtrage d'interfrégramme et le déroulement de phase pour qu'ils soient adaptatifs aux problèmes pouvant être rencontrés.

Pour cela, notre thèse porte sur quatre chapitres. Dans le premier chapitre, nous allons préparer le terrain en donnant certaines notions de base sur le système d'imagerie inSAR, les bandes de fréquences utilisées, le fonctionnement, caractéristiques et les différentes images générées. Ce chapitre permettra également de résumer les mécanismes de génération et les caractéristiques des interférogrammes dédiés aux topographies du terrain.

Le deuxième chapitre fait l'objet d'un état de l'art sur les différents algorithmes de déroulement [19-45], plusieurs approches de déroulement adaptées à la présence des résidus ont été proposées, elles se divisent en deux catégories principales : suivi de parcours (*PathFollowing*) et norme de minimisation (*Minimum Norm*).

La catégorie PF se divise elle-même en deux sous-classes : rupture des branches (*BranchCut*) [20-24] et guidée par la qualité-croissance des régions (*QualityGuided- regionGrowing*) [29-36]. Le traitement PF se caractérise par des techniques locales et il déroule les phases les unes après les autres en suivant un parcours statique prédéterminé ou dynamique adaptatif à localisation des résidus (des branches ou des grappes). Tandis que la catégorie MN [36-45] se caractérise par un traitement global dont toutes les phases sont impliquées à la fois dans le calcul de la solution, nous nous trouvons dans ce cas face à l'utilisation des méthodes numériques pour résoudre le problème. Dans ce chapitre nous allons entamer cette étude par l'utilité de déroulement avec ses formulations de base, dans la deuxième section une description mathématique détaillée de déroulement sera présentée et par cette description une méthode d'ajustement sera ainsi déduite [17]. En suite la dernière section consiste à un état de l'art sur des méthodes de déroulement largement utilisées dans les deux catégories, cependant une tendance vers les approches PF sera adoptée. Le troisième chapitre examine en détail les différents codes correcteurs d'erreurs, en se concentrant sur les codes de Reed-solomon. Les derniers seront bien détaillés dans leur aspect mathématique, codage et décodage en donnant des exemples pour mieux éclaircir leurs stratégies, principes et techniques. Nous démontrons dans le quatrième chapitre que le code de RS connu est à la fois un code de chiffrement et de codage de canal. La méthode appliquée dépend de la minimisation des attaques et des erreurs après la transmission de l'interférogramme INSAR crypté. Les résultats étaient satisfaisants.

# **Chapitre I**

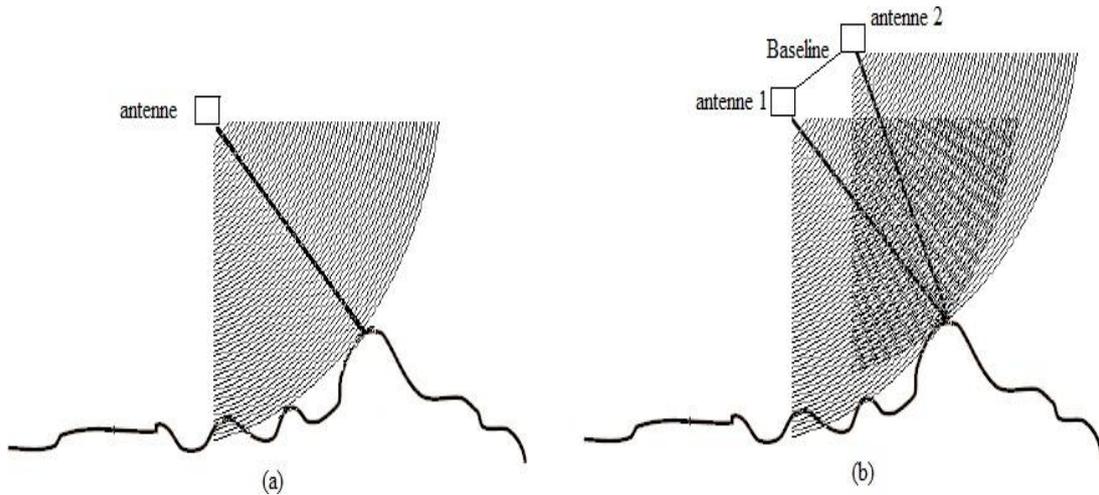
## **Généralité sur le système d'imagerie inSAR**

## I.1 Introduction

Dans ce chapitre, nous allons préparer le terrain en donnant certaines notions de base sur le système d'imagerie inSAR, les bandes de fréquences utilisées, le fonctionnement, caractéristiques et les différentes images générées. L'avantage de ce système comparativement à celui optique est sa capacité de fonctionner le jour et la nuit et dans toutes les conditions météorologiques car il possède sa propre source d'illumination qui est l'onde EM. Les domaines d'application d'un tel système est très vaste telles que : la météorologie, changement des surfaces, suivi le mouvement des terrains, la fonte des glaciers et les inlandsis, activités et éruptions volcaniques et la topographie [1-3]. Cette dernière consiste le point d'intérêt de notre travail, la topographie est obtenue par le processus de génération de modèle numérique du terrain (*DEM Digital Earth Model*) en exploitant les images de phases de l'inSAR communément nommées les interférogrammes. Donc ce chapitre permettra de résumer les mécanismes de génération et les caractéristiques des interférogrammes dédiés aux topographies du terrain.

## I.2 C'est quoi le système inSAR

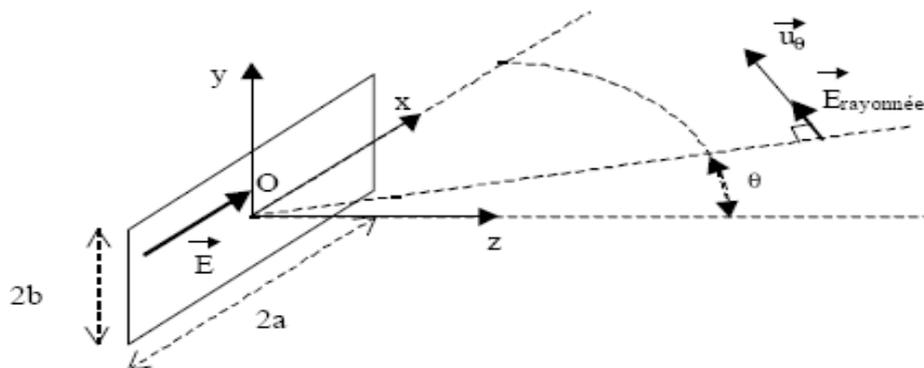
Nous pouvons dire que le système inSAR est chaque méthode employant deux signaux complexes issus de deux images SAR [4]. En effet le système inSAR se diffère du SAR par une double acquisition de signaux collectés par une seule antenne et multi-passe ou par deux antennes et mono-passe, dont le premier est utilisé pour détecter les changements (décorrélation temporelle) et le second est pour les informations topographiques. Alors le système inSAR est principalement constitué une ou deux antennes séparées par une baseline, elles sont spatiaux-portés ou aéroportées, le signal capté est due à la corrélation entre les deux signaux de deux antennes, l'un d'eux est dit maître et l'autre est esclave. Par l'antenne SAR la seule information exploitable est l'amplitude car la phase est inutile et n'a pas de relation avec l'altitude de point imagé, ce qui le rend un système générant les images d'amplitude. Par contre le système inSAR ; plus l'amplitude ; peut fournir une information de phase liée à l'altitude de point imagé grâce l'interférence entre les deux signaux réfléchis.



**Figure I.1** : Différence entre SAR et inSAR, (a) SAR, (b) inSAR

### I.2.1 L'antenne radar

Un conducteur parcouru par un courant électrique alternatif produit un champ électromagnétique qui rayonne dans l'espace environnant, pour une bonne directivité le champ EM se met à propager dans un guide d'onde. Puis se termine à rayonner via une ouverture généralement rectangulaire pour les systèmes inSAR.



**Figure I.2** : L'ouverture rayonnante rectangulaire.

Le champ max est obtenu par l'angle  $\theta=0$ , la forme d'illumination au sol (une section de diagramme de rayonnement) est elliptique.

### I.2.2 Bandes de fréquences

Le tableau I.1 ci-dessous montre les noms, les bandes de fréquences, les longueurs d'onde et leurs domaines d'application.

Le nom de la bande	Les fréquences	Les longueurs d'onde	utilisations
P	<400Mhz	<1m	Pénétration, humidité du sol, la biomasse.
L	1-2 Ghz	15-30 cm	Végétation, déformation
S	2-4 GHz	7.5-15 cm	Végétation, déformation
C	4-8 Ghz	3.75-7.5 cm	Agriculture, déformation, topographie
X	8-12 Ghz	2.5-3.75 cm	Topographie haute résolution
Ku	12-18 Ghz	1.67-2.5 cm	Les océans les larges surface d'eau
Ka	24-40Ghz	0.75-1.11 cm	Glace et neige
Mm	40-300 Ghz	0.75-1 cm	Glace et neige

**Tableau I.1** : Les bandes de fréquences SAR et inSAR. [3]

### I.2.3 Quelques satellites SAR et inSAR

Le tableau I.2 Ci-dessous montre quelques satellites spécifiques.

Nom	Société propriétaire	Longueur d'onde	Date d'opération
US Seasat	National Aeronautics and SpaceAdministration (NASA)	Bande-L k = 23.5 cm	Juin-octobre 1978
EuropeanRemote-sensing Satellite 1 (ERS-1)	EuropeanSpace Agency (ESA)	Bande -C k = 5.66 cm	1991–2000
JapaneseEarthResources Satellite 1 (JERS-1)	JapanAerospace Exploration Agency (JAXA)	Bande- L k = 23.5 cm	1992–1998

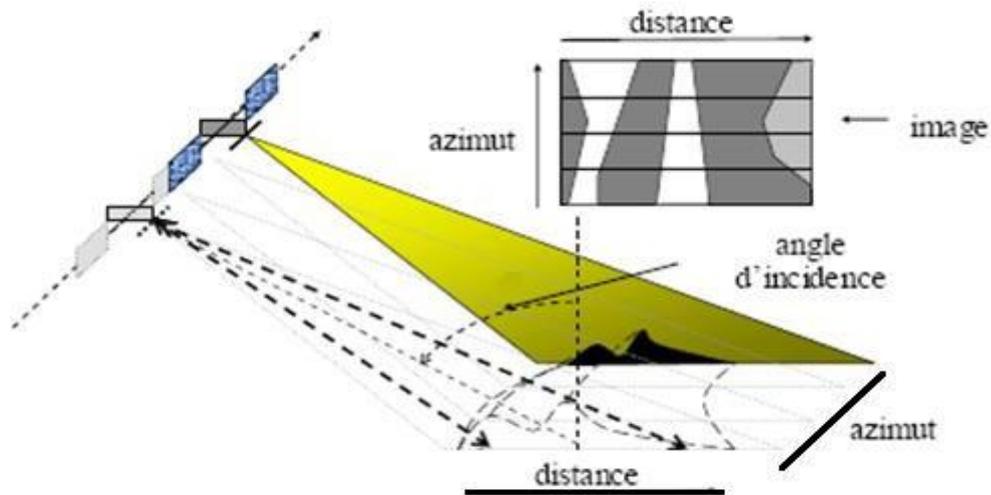
European Remote-sensing Satellite 2 (ERS-2)	European Space Agency (ESA)	Bande- C k = 5.66 cm	1995–2011
Canadian Radar Satellite 1 (Radarsat-1)	Canadian Space Agency (CSA)	Bande- C k = 5.66 cm	1995–2013
European Environmental Satellite (Envisat)	European Space Agency (ESA)	Bande- C k = 5.63 cm	2002–2012
Japanese Advanced Land Observing Satellite (ALOS)	Japanese Aerospace Exploration Agency (JAXA)	bande -L k = 23.6 cm	2006–2012
Canadian Radar Satellite 2 (Radarsat-2)	Canadian Space Agency (CSA)	Bande- C k = 5.55 cm	2007–présent
TerraSAR-X	German Aerospace Center (DLR)	bande -X k = 3.1 cm	2007–présent
COSMO-SkyMed constellation	Italian Space Agency (ASI)	Bande- X k = 3.1 cm	2007- présent
TerraSAR Add-on for Digital Elevation Measurements (TanDEM-X)	German Aerospace Center (DLR)	Bande- X k = 3.1 cm	2010- présent

**Tableau I.2** : Quelques satellites SAR [5].

### I.3 Principe de SAR

#### I.3.1 la résolution

Les anciens radars d'imagerie à vision latérale (SLAR) basés sur une antenne à ouverture réelle envoient des impulsions avec basses fréquences; ici il ne s'agit pas la longueur d'onde ni la durée d'impulsion mais il s'agit le temps séparant deux impulsions; lors de la réception de l'écho et la retransmission de deuxième impulsion le porteur pourrait déjà être endors de la cellule illuminée précédente (pas de chevauchement), avec les systèmes SAR certaines améliorations ont pu être obtenues au niveau de la résolution en introduisant la technique de l'antenne virtuelle synthétique c.-à-d. une antenne composée de plusieurs cellules synthétisées due à la haute fréquence des impulsions.



**Figure I.3** : Notion d'illumination en azimute et en distance.

Ceci porte une bonne amélioration de la résolution en azimute car il y a une augmentation virtuelle de la taille d'antenne, tandis que la résolution en distance reste dépendante de la durée de l'impulsion.

#### La résolution en distance

La résolution en distance peut être définie comme la distance la plus courte pour laquelle deux cibles ponctuelles produisent deux échos non-chevauchés. La figure I.4 montre qu'une impulsion de durée  $\tau$  donne une résolution distale :

$$Rd = \frac{c\tau}{2} \quad (I.1)$$

Avec  $c$  est la célérité.

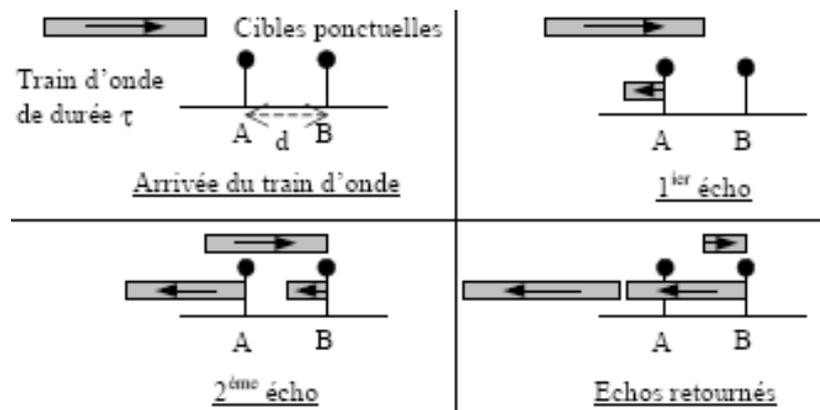
Donc une bonne (fine) résolution distale exige une impulsion très brève, ce qui présente une contrainte technologique et économique. Pour surmonter ce problème il existe en traitement de signal une technique dite compression d'impulsion (*Chirp pulse Compression*) qui consiste à transmettre longue impulsion, puis compresser tous les échos [6].

La véritable résolution distale au sol n'est qu'une projection selon l'angle d'incidence de faisceau  $\theta$ , l'équation I.1 devient :

$$Rd_{sol} = \frac{c\tau}{2 \sin(\theta)} \quad (I.2)$$

Autre approximation est proposée dans [6-8] en développant la formule de telle façon qu'elle soit en fonction de la bande passante de l'impulsion B.

$$Rd'_{sol} = \frac{c}{2 B \sin(\theta)} \quad (I.3)$$



**Figure I.4** : La durée d'impulsion et la résolution distale.

#### La résolution en azimuth

La résolution azimutale dépend principalement de dimension d'antenne ou son ouverture rayonnante [9], elle peut être définie comme :

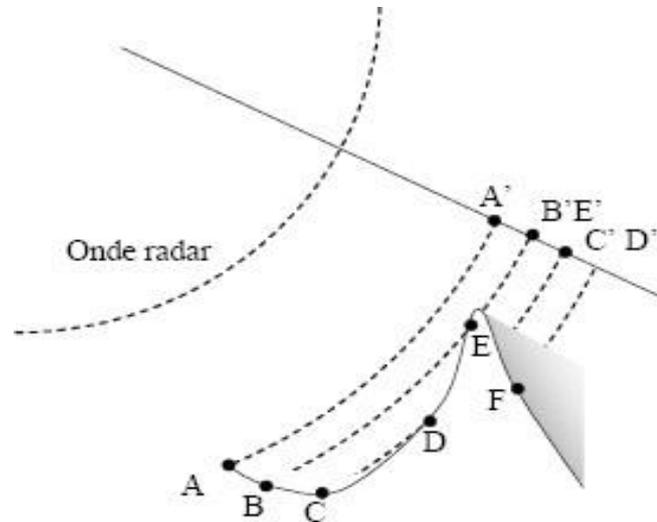
$$Ra = \frac{\lambda r}{D} \quad (I.4)$$

Où  $\lambda$  est la longueur d'onde,  $r$  la distance entre l'antenne et le point imagé et  $D$  est l'ouverture d'antenne [7,8]. Cette expression de  $Ra$  veut dire que; pour une longueur d'onde et distance  $r$  données; nous pouvons avoir une bonne (fine) résolution azimutale avec antenne de grande ouverture (dimension), par exemple pour une  $Ra$  désirée de 10 m, et une distance typique de porteur de 800 km tel que le satellite ERS-1/2 qui fonctionne dans la bande C de  $\lambda=5.66$  cm (voir les tableaux I.1 et 2) nous devons avoir une ouverture

D=448m, une telle dimension était auparavant impossible de se placer sur aucun satellite SLAR. Avec le SAR nous pouvons surmonter la limitation de dimension réelle d'antenne en la remplaçant par une autre virtuelle synthétique, cette technique a été introduite la première fois pour des applications militaires par Carl Wiley en 1951 (Goodyear Aircraft Corporation).

### I.3.2 les défauts de projection

La projection parabolique de l'onde EM sur un relief peut causer des défauts dues à la courbure et à l'inclinaison des quelques points imagés comme il est montré dans la figure I.5 ci-dessous, ce phénomène est l'une des sources de bruits dans les images SAR.



**Figure I.5 :** Les défauts de projection parabolique de l'onde radar.

#### Zones de compression ou zones d'accumulation

C'est le cas des points B et E où leurs échos se font simultanément car ils se situent sur la même distance par rapport à l'antenne, alors B' et E' se voient accumulés. Ces points se traduisent dans l'image par des pixels de forte brillance (blanc).

#### Zones d'inversion

C'est le cas du point E par rapport aux C et D où l'écho de E se reçoit le premier avant ceux de C et D, il constitue un cas extrême de zone de compression, ce défaut provient principalement des points de forte pente situant au sommet du relief.

### Zones d'ombre

C'est le cas de point F où il y aura aucun écho car ce point se situe dans l'ombre, ce défaut est similaire aux systèmes d'imagerie optique sauf que l'ombre de SAR provient de l'onde EM alors que celui de système optique dépend de la position du soleil et aussi la présence des nuages. Ces points se traduisent dans l'image par des pixels noirs.

### **I.3.3 le signal et les images SAR**

Le signal collecté pour une cellule de résolution est complexe possédant une amplitude par laquelle nous obtiendrons l'image d'amplitude SAR et une phase qui est inutile pour générer une image ce qui n'est pas le cas de l'inSAR où les deux informations peuvent être exploitées afin d'avoir deux types d'image.

Dans les images d'amplitude la valeur de chaque pixel dépend des caractéristiques de réflectivité ou de la rétrodiffusion du point imagé et aussi l'orientation de la surface cellule. Pourtant l'information d'amplitude est directement exploitable et techniquement facile à extraire mais malheureusement elle fonction de nombreux paramètres de rétrodiffusion et géométriques ce qui rend l'interpolation une opération complexe ou dans certain cas basé sur des méthodes empiriques pour modéliser les valeurs par un niveau de brillance pondéré et ensuite une série des corrections seront aussi nécessaires [6, 10].

Parmi ces paramètres nous en trouvons :

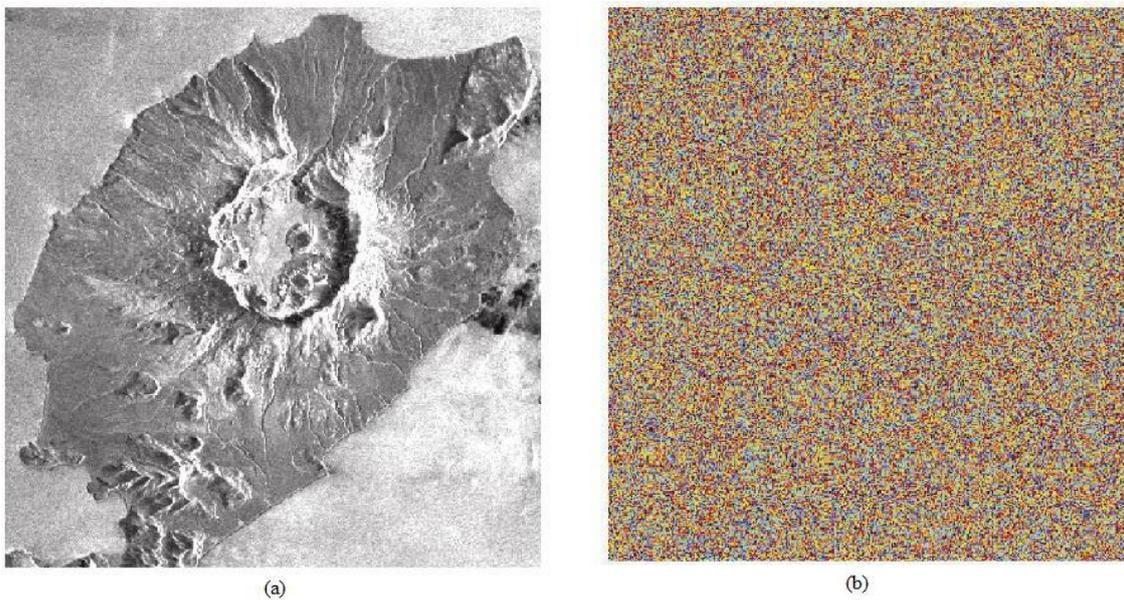
- Absorption par la cible.
- Diffusion par la cible selon un facteur dit la radiométrie.
- Réflexion dans des directions autres que celle de l'antenne.
- Le facteur d'inclinaison.
- Le facteur d'orientation.

L'image de phase SAR est inutile car elle ne donne pas une information sur la brillance et elle est composée de plusieurs phases ce qui rend l'image comme une distribution aléatoire et sans aucun sens (voir la figure I.6.b)

$$\varphi = \varphi_{\text{trajet}} + \varphi_{\text{atmosphérique}} + \varphi_{\text{rétrodiffusion}} + \varphi_{\text{instrumentale}} + \varphi_{\text{bruit}} \quad (\text{I.5})$$

- La phase de trajet est le déphasage lié à la distance parcouru par l'onde.

- La phase météorologique est un second déphasage de trajet par des conditions atmosphériques particulières (dû à l'interaction de l'onde avec les particules météorologiques).
- La phase de rétrodiffusion est le déphasage introduit au moment d'interaction onde-surface.
- La phase instrumentale est celle produite par le système électronique d'émission et de réception (temps de réponse des filtres, amplificateurs...).
- Un bruit de phase dû aux procédés d'acquisition (numérisation, dérives fréquentielles de la porteuse...).



**Figure I.6 :** Images SAR, (a) image d'amplitude, (b) image de phase.

#### I.4 Principe de l'inSAR

L'inSAR se diffère de SAR par une double acquisition de signal par deux antennes pour le mode mono-passe ou deux acquisitions par même antenne pour le mode multi-passe. La figure I.7 présente un système inSAR mono-passe comporte deux antennes séparées par une ligne de base (*baseline*)  $B_x$ , une dite maîtresse et l'autre dite esclave. Le but de tel système est d'analyser les différences de phases entre les deux signaux captés, cette analyse permet d'ajouter un appui à l'information d'amplitude qui reste la même en SAR ou inSAR.

Il est clair que la différence de phase  $\Delta\varphi$  due à la différence de trajet  $\Delta r$  entre les deux signaux est :

$$\Delta\varphi = \frac{4\pi}{\lambda} \Delta r \quad (I.6)$$

D'autre part, géométriquement nous avons :

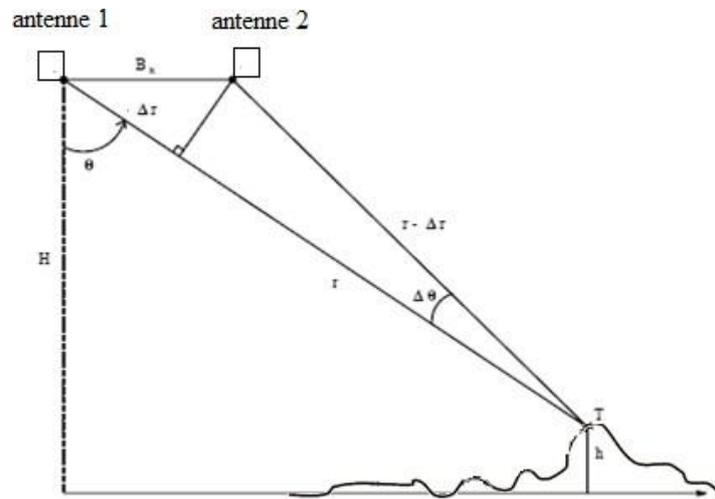
$$h = H - r \cos \left[ \arcsin \left( \frac{\Delta r}{B_x} \right) \right] \quad (I.7)$$

D'après les équations I.6 et 7 nous pouvons directement déduire :

$$\Delta\varphi = \frac{4\pi B_x}{\lambda} \sin \left( \arccos \left( \frac{H-h}{r} \right) \right) \quad (I.8)$$

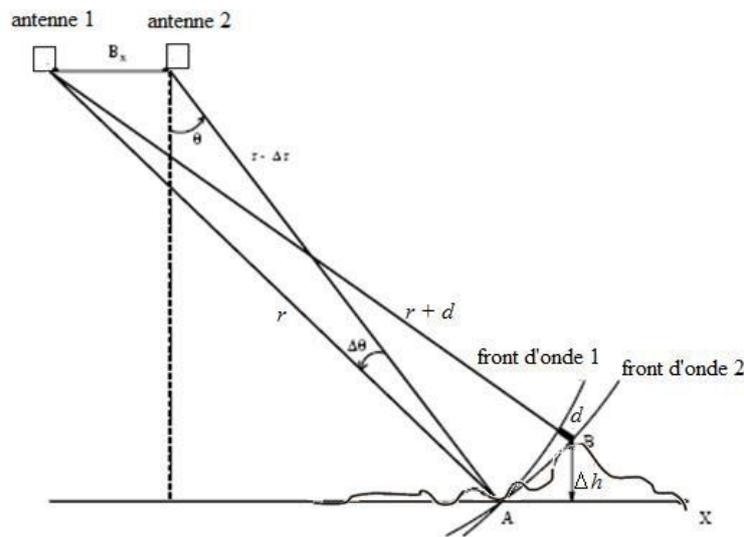
L'équation I.8 est la proportionnalité entre l'altitude de point imagé  $h$  et la différence de phase de trajet de deux signaux  $\Delta\varphi$  qui sera dorénavant nommée la phase interférométrique  $\phi$  car elle est due de l'interférence entre deux signaux corrélés.

Donc cette technique d'interférométrie permet d'avoir une relation directe entre l'altitude est la phase interférométrique mesurée par la corrélation (voir paragraphe I.5) et cette phase mesurée ne dépend que de la différence de trajet dont toutes les autres phases de l'équation I.5 seront éliminées par soustraction à part le bruit parce que les deux signaux captés se propagerait dans les mêmes conditions.



**Figure I.7** : L'altitude de point imagé et la différence de trajet.

Malheureusement la phase mesurée est modulo  $2\pi$  et par conséquent l'altitude est roulée à une hauteur proportionnelle à  $2\pi$  dite l'altitude d'ambiguïté  $h_a$  qui est définie comme une différence d'altitude donnant une différence de phase interférométrique égale  $2\pi$  et cela correspond à une différence de trajet de  $\lambda/2$ . À ce stade nous avons élaboré deux notions de différence de phase, la première est la différence de phase de trajet entre les deux signaux corrélés  $\Delta\phi$  proportionnelle au  $\Delta r$  pour un point, la seconde est la différence de phase interférométrique entre deux points  $\Delta\phi = \Delta(\Delta\phi)$  proportionnelle à la différence de distance radiale  $d$  de l'antenne maîtresse (figure I.8).



**Figure I.8** : La différence d'altitude et la différence de la distance radiale.

La figure I.8 simplifie l'analyse de différence d'altitude  $\Delta h$  entre deux points A et B en fonction de la distance radiale  $d$  et par conséquent en fonction de la phase interférométrique.

$$\Delta h = \frac{d r \tan \theta}{Bx} \quad (I.9)$$

Nous pouvons également définir l'altitude d'ambiguïté comme :

$$h_a = \frac{\lambda r \tan \theta}{2Bx} \quad (I.10)$$

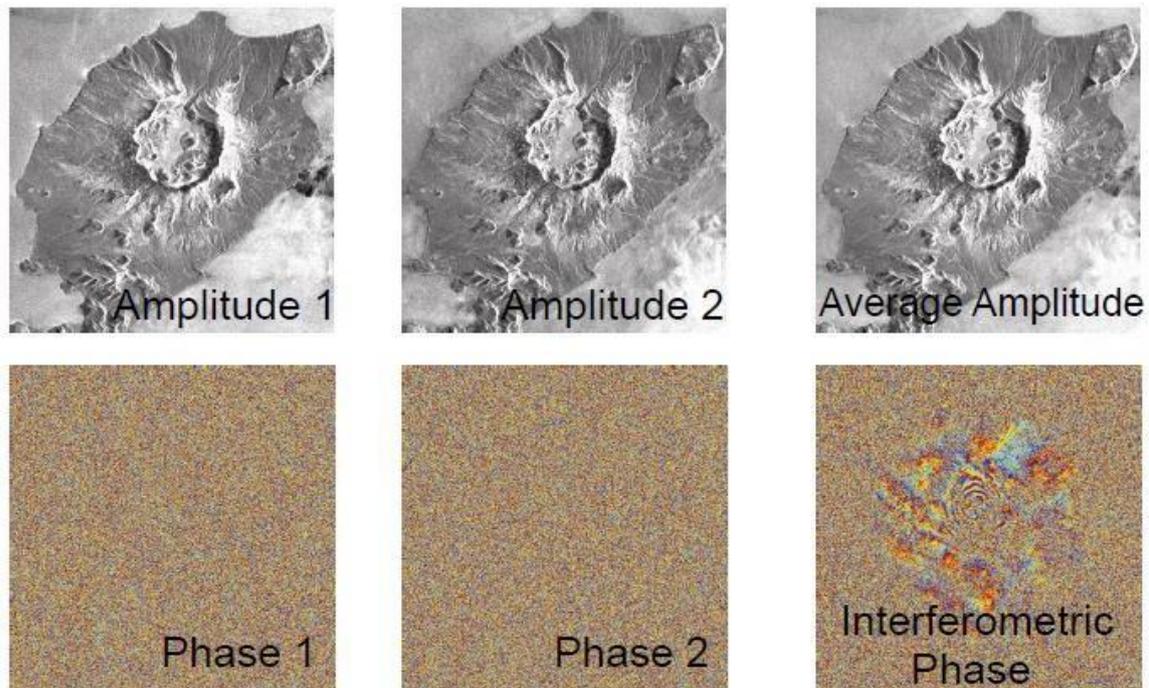
Cette notion d'amplitude d'ambiguïté se manifeste comme limite des franges dans l'image de phase interférométriques communément appelée interférogramme, alors le nombre de franges augmente proportionnellement avec la fréquence et la *baseline*, et inversement proportionnelle avec la hauteur de l'orbite (altitude du porteur).

Exemple : ERS-1,  $\lambda=5.66$  cm,  $r = 800$ km,  $\theta=23$ ,  $Bx = 1300$ m ce qui donne un  $h_a = 7.2$ m

Nous pouvons aussi définir une autre limite concernant la taille de la cellule de résolution (taille de pixel) où la distance radiale  $dc$  critique ne doit pas la dépasser, cette  $dc$  critique se traduit par une altitude dite de superposition  $h_s$ , qui induit à une déformation à l'échelle de la taille de la cellule de résolution  $Rd$  (équation I.1,2).

$$h_s = \frac{Rd r \tan \theta}{Bx} \quad (I.11)$$

La figure I.9 montre un exemple de l'image d'amplitude, les images de phase singulière et l'interférogramme générés par l'inSAR.



**Figure I.9 :** Images inSAR.

### I.5 Signal de corrélation et les images inSAR

Les deux signaux captés par les antennes sont ensuite combinés par le produit de corrélation qui est une multiplication entre le signal de l'antenne maîtresse  $S_1$  et le conjugué de l'autre  $S_2^*$  d'où vient l'élimination des phases communes. En traitement de signal le produit de corrélation entre deux signaux est mentionné par :

$$P_{\text{corrélation}} = \langle S_1 \cdot S_2^* \rangle \quad (\text{I.12})$$

La stabilité de prise de l'image est évaluée par un coefficient dit de corrélation de système  $\rho_{\text{système}}$  par lequel nous pouvons juger que le signal au niveau de la cellule de résolution originale est corrélé ou décorrélé.

$$\rho_{\text{système}} = \frac{\langle S_1 \cdot S_2^* \rangle}{\sqrt{\langle S_1 \cdot S_1^* \rangle \langle S_2 \cdot S_2^* \rangle}} \quad (\text{I.13})$$

Le coefficient de corrélation réel n'est pas seulement de système mais il est accompagné par deux termes de décorrélations spatiale  $\rho_{\text{spatiale}}$  et temporelle  $\rho_{\text{temporelle}}$ .

$$\rho_{\text{totale}} = \rho_{\text{système}} \rho_{\text{spatiale}} \rho_{\text{temporelle}} \quad (\text{I.14})$$

Une valeur 1 de ce coefficient indique une forte corrélation (max), et 0 indique une décorrélations totale (min).

### I.5.1 Décorrélations spatiales

La *baseline* séparant les deux antennes est la source de ce phénomène parce que la décorrélations spatiales est due à l'observation sous deux angles d'incidence différents de la même cellule de résolution [11] :

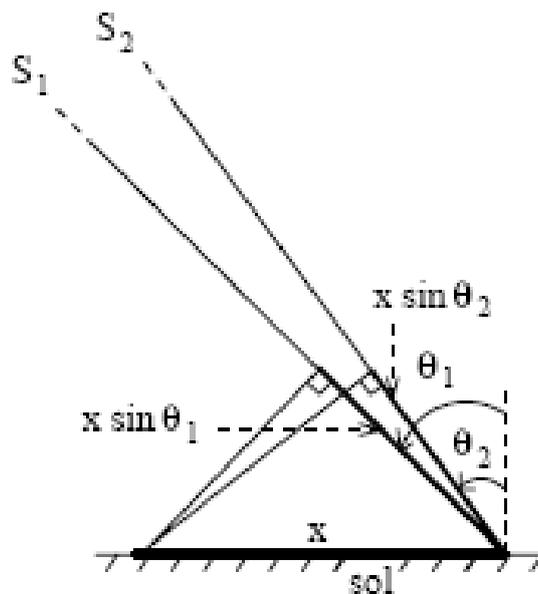


Figure I.10 : Les angles d'incidence et la décorrélations spatiales.

Le coefficient de décorrélation spatial vaut [11] :

$$\rho_{spatiale} = 1 - \frac{2rd \Delta\theta \cos(\theta_2)}{\lambda} \quad (I.15)$$

Avec  $\Delta\theta = \theta_1 - \theta_2$

Il est évident que :  $r_2 \Delta\theta = Bx \cos \theta_2$  avec  $r_2$  est la distance qui sépare le point de l'antenne esclave 2 ( $r - \Delta r$  dans la figure I.7 et 8).

Alors l'équation I.15 en fonction de la *Baseline* Bx est:

$$\rho_{spatiale} = 1 - \frac{2rd Bx (\cos(\theta_2))^2}{\lambda r_2} \quad (I.16)$$

### I.5.2 Décorrélation temporelle

L'observation d'une région dans deux temps différents est la source de ce phénomène, la décorrélation temporelle se résulte de changement de l'état de la surface entre les deux observations ce qui le rend propre au mode multi-passe. Pour les applications de détection de changements et de mouvement telles que : le gisement, le suivi de déplacement et la fonte des glaciers, les éruptions volcaniques et les activités sismiques, la décorrélation temporelle est très utile et elle présente le seul critère de détection. Mais pour le MNT les informations d'altitude doivent être inchangées entre les deux observations c'est pour cela que le mode mono-passe lui est approprié.

Zebker [12] a estimé que ces changements à l'intérieure d'une cellule de résolution est une fonction de probabilité gaussienne possédant un écart-type horizontal  $\sigma_h$  et vertical  $\sigma_v$ .

Le coefficient de décorrélation temporelle vaut :

$$\rho_{temporelle} = \exp \left[ -\frac{1}{2} \left( \frac{4\pi}{\lambda} \right)^2 ((\sigma_h \sin\theta)^2 + (\sigma_v \cos\theta)^2) \right] \quad (I.17)$$

Avec  $\theta$  est l'angle d'incidence de l'antenne maîtresse.

### I.5.3 les images inSAR

En interférométrie, pour générer les différentes images inSAR un autre coefficient complexe  $\gamma$  déduit du coefficient de corrélation  $\rho$  est utilisé. Il consiste à faire la moyenne

(sous échantillonnage) d'un ensemble des pixels (cellules de résolution originale) dans une fenêtre, il est comme une sorte de passage de haute résolution à basse résolution dans le but de préfiltrer les bruits, minimiser la taille immense des images et faire une correction géométrique (recalage) de la décorrélation spatiale.

$$\gamma(i',j') = \frac{\sum_{ij}^{MN} S_1(i,j)S_2^*(i,j)}{\sqrt{\sum_{ij}^{MN} S_1(i,j)S_1^*(i,j) \cdot \sum_{ij}^{MN} S_2(i,j)S_2^*(i,j)}} \quad (\text{I. 18})$$

Avec :  $(i',j')$  les indices des pixels de la nouvelle résolution,  $(i,j)$  ceux de la résolution originale,  $(M,N)$  dimension de l'image.

Trois types d'images pouvant être générées :

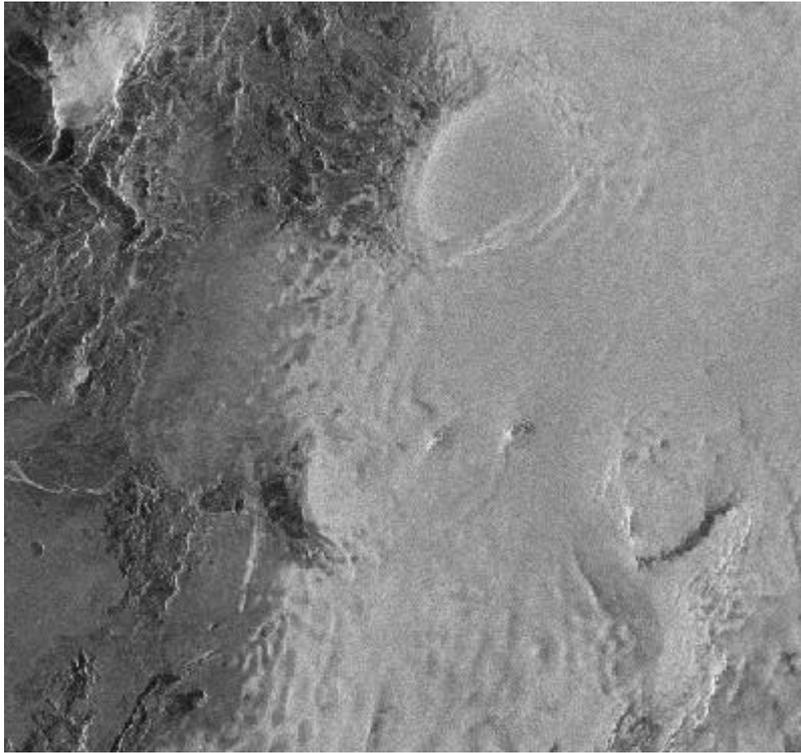
- L'image d'amplitude : qui est la simple moyenne de la fenêtre de basse résolution

$$A(i',j') = \frac{1}{MN} \left| \sum_{ij}^{MN} S_1(i,j)S_2^*(i,j) \right| \quad (\text{I. 19})$$

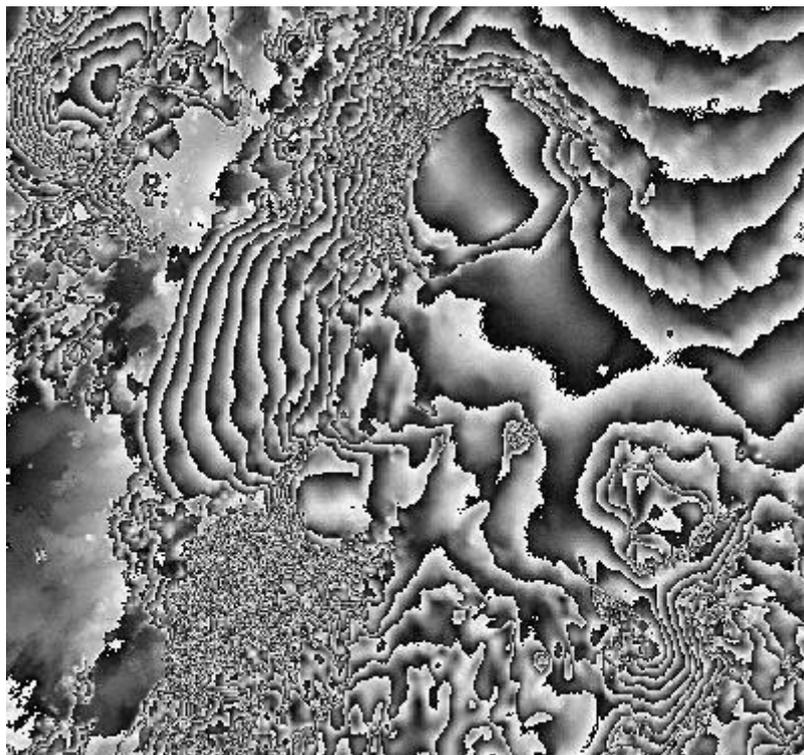
Elle se diffère de celle de SAR seulement par le moyennage, c'est pour cela qu'il est mentionné dans la figure I.9 « *average amplitude* ».

- L'image de phase interférométrique (interférogramme) : qui est la phase de  $\gamma(i',j')$ .
- L'image de cohérence : qui est le module de  $\gamma(i',j')$ , elle est considérée comme une image indiquant la qualité de mesure et largement utilisée pour le filtrage ou le déroulement de phase.

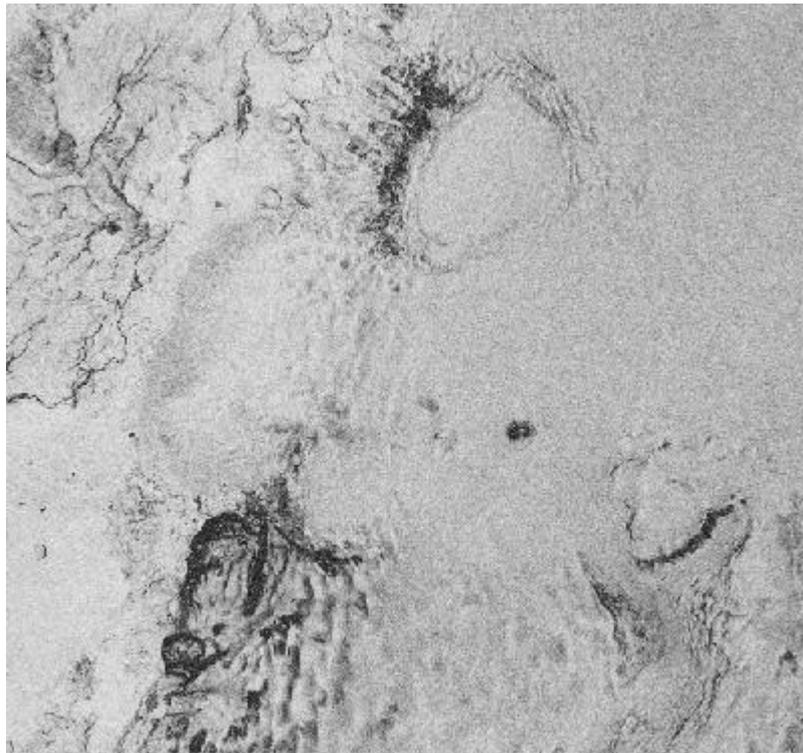
Les figures I.11 (a, b, c) montrent des exemples réels sur les différentes images inSAR d'ERS-1/2 (*tandem data*) fournies par ESA, il s'agit d'une région de Vatnajökull Islande, dimension 376 x 400 pixels, prise le 31 décembre 1995 sur l'orbite 23315.



(a)



(b)



(c)

**Figure I.11 :** Exemple d'image inSAR de Vatnajökull ERS-1/2 (*tandem data*) ESA, (a) image d'amplitude, (b) interférogramme, (c) image de cohérence.

## I.6 Conclusion

Dans ce chapitre nous avons essayé de donner une brève étude pour comprendre le principe et les composants essentiels des systèmes d'imagerie radar destinés aux applications topographiques du terrain, ce qui permet d'avoir une idée générale sur les différents types d'images pouvant être générées par un tel système et leurs caractéristiques.

Notamment pour les images de phase interférométriques (interférogrammes), cette étude est considérée comme un *Background* suffisant avant d'entamer le filtrage et le déroulement qui constituent le but de notre travail et qui font l'objet des prochains chapitres.

# **Chapitre II**

## **Étude, analyse et comparaison de différentes méthodes de déroulement de phase**

## II.1 Introduction

Après le filtrage d'interférogramme, l'étape suivante de la chaîne de traitement d'image inSAR est le déroulement de phase. Cette information est mesurée dans l'intervalle  $[-\pi, \pi]$  due à l'opération de roulement d'où vient le nom la phase roulée (*Wrapped-Phase*), et tant que l'obtention de l'altitude exacte du point imagé est proportionnelle à la phase réelle absolue (voir chapitre I) cette phase doit être reconstruite par une opération inverse non linéaire pour avoir une phase déroulée (*Unwrapped -phase*).

Si l'interférogramme est sans bruits (sans aucun résidu); ce qui est impossible en pratique, le déroulement devient une procédure très facile et plusieurs solutions sont possibles selon le pixel de départ et le parcours d'intégration suivi. Mais avec la présence des résidus l'application directe d'une telle procédure échoue en engendrant des lignes distordues le long du parcours si l'intégration passe via ces résidus. Pour remédier à ce problème plusieurs approches de déroulement adaptées à la présence des résidus ont été proposées, elles se divisent en deux catégories principales : suivi de parcours (*Path-Following*) et norme de minimisation (*Minimum-Norm*).

La catégorie PF se divise elle-même en deux sous-classes : rupture des branches (*Branch-Cut*) et guidée par la qualité -croissance des régions (*Quality-Guided- region-Growing*). Le traitement PF se caractérise par des techniques locales et il déroule les phases une après autre en suivant un parcours statique prédéterminé ou dynamique adaptatif à localisation des résidus (des branches ou des grappes).

Tandis que la catégorie MN se caractérise par un traitement global dont toutes les phases sont impliquées à la fois dans le calcul de la solution, nous nous trouvons dans ce cas face à l'utilisation des méthodes numériques pour résoudre le problème.

Dans ce chapitre nous allons entamer cette étude par l'utilité de déroulement avec ses formulations de base, dans la deuxième section une description mathématique détaillée de déroulement sera présentée, et par cette description une méthode d'ajustement sera ainsi déduite. En suite la dernière section consiste à un état de l'art sur des méthodes de déroulement largement utilisées dans les deux catégories, cependant une tendance vers les approches PF sera adoptée car le travail personnel en fait partie.

## II.2 C'est quoi le déroulement de phase ?

### II.2.1 formulations

Nous avons vu précédemment que la phase interférométrique obtenue par le système inSAR a une proportionnalité directe avec la hauteur de point imagé, mais seulement la phase réelle qui peut interpréter cette altitude réelle et comme nous savons que la phase est mesurée dans l'intervalle  $[-\pi, +\pi]$  cette altitude sera aussi en ambiguïté. Autrement dit, seulement la valeur principale, qui est définie comme le reste de la phase après la soustraction le maximum de multiple  $2\pi$ , sera détectée. Donc la procédure de déroulement de phase consiste à reconstruire la phase réelle à partir de celle roulée en trouvant le nombre de cycle exacte à ajouter.

$$\psi_{i,j} = w(\phi_{i,j}) = \phi_{i,j} + 2k\pi \quad (\text{II. 1})$$

Avec

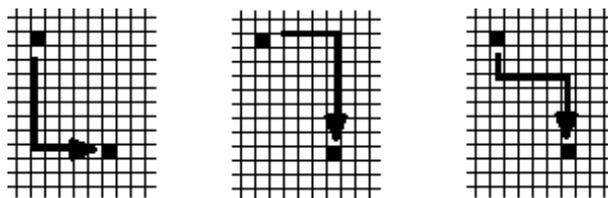
$$w(\phi_{i,j}) = \text{mod}\{[\pi + \phi_{i,j}], \quad 2\pi\} - \pi \quad (\text{II. 2})$$

Et  $k \in \mathbb{Z}$

Cette opération est non-linéaire et elle possède les propriétés suivantes :

$$\begin{aligned} f &= w^{-1}wf \\ wf &= wwf \\ w^{-1}f &= w^{-1}w^{-1}f \\ \sin(f) &= \sin(wf) \end{aligned}$$

Le déroulement de phase est un problème, s'il n'y a pas d'autres informations supplémentaires telle qu'une altitude et coordonnées exactes de point de départ. En effet, il existe une infinité de possibilités correspondant aux images de phase déroulées pour une seule image roulée. Cette lacune d'unicité d'image déroulée est montrée dans l'exemple ci-dessous.



**Figure II.1** : Problème non-unicité de déroulement selon le parcours suivi et le point de départ

La figure II.1 montre un exemple de déroulement pour une petite portion d'image roulée où nous remarquons que nous pouvons trouver plusieurs solutions possibles de déroulement selon le pixel de départ et les parcours empruntés.

La formule qui permet de calculer la phase déroulée  $\Psi_i$  à partir de l'intégration de gradient roulé discret accumulé de la phase roulée  $\phi_n$  unidimensionnelle est

$$\Psi_i = \phi_1 + \sum_{n=1}^{i-1} w(\Delta\phi_n) \quad (\text{II. 3})$$

Où  $\Delta\phi_n$  est le gradient discret roulé de la phase roulée. il s'écrit :

$$\Delta\phi_n = w(\phi_{n+1} - \phi_n) \quad (\text{II. 4})$$

Pour le traitement bidimensionnel la phase déroulée peut facilement être calculée par l'intégration de gradient accumulé bidimensionnelle suivant :

$$\Psi_{i,j} = \phi_{1,1} + \sum_{n=1}^{i-1} w(\Delta_1\phi_{n,1}) + \sum_{m=1}^{j-1} w(\Delta_2\phi_{i,m}) \quad (\text{II. 5})$$

Avec  $\Delta_1\phi_{n,1} = w(\phi_{n+1,1} - \phi_{n,1})$  est le gradient vertical et  $\Delta_2\phi_{i,m} = w(\phi_{i,m+1} - \phi_{i,m})$  est le gradient horizontal. Alors les solutions possibles de  $\Psi_{i,j}$  dépendent de pixel de départ  $\phi_{1,1}$ ; qui peut être différent de (1,1), et le parcours suivi par l'intégration.

Dans le cas de présence de bruits (résidus) l'application de ce cumule des gradients roulés engendre des lignes distordues le long de parcours (chemin) d'intégration choisi, ce que nous appelons une dépendance de parcours (*Path-depending*). Avant d'entamer les algorithmes de déroulement résolvant le problème de dépendance de parcours nous essayons de donner une description mathématique avec une petite amélioration visant à minimiser légèrement les zones distordues ou évitées lors de déroulement.

### II.2.2 La condition d'Itoh

Pour que le déroulement soit correctement exécuté sans aucune confusion entre les discontinuités dues aux franges (discontinuités de l'opération modulo  $2\pi$ ) et les discontinuités de mauvais échantillonnage une condition très d'importante est adoptée par toutes les méthodes

de déroulement, elle consiste à exiger que toutes les variations en valeur absolue entre deux pixels adjacents sont inférieures à  $\pi$ . Elle est communément appelée la condition d'Itoh [13]. Nous considérons le cas unidimensionnel d'une ligne d'image de phase réelle ou déroulée.

$$\Delta\Psi_i = \Psi_i - \Psi_{i-1} \quad (\text{II. 6})$$

La condition d'Itoh est :

$$|\Delta\Psi_i| \leq \pi \quad (\text{II. 7})$$

Et à partir de II.6 il est évident que :

$$\sum_{i=1}^N \Delta\Psi_i = \Psi_N - \Psi_0 \quad (\text{II. 8})$$

À partir de l'équation II.1 nous avons :

$$\Delta w(\Psi_i) = \Psi_i - \Psi_{i-1} - 2\pi(k_i - k_{i-1}) \quad (\text{II. 9})$$

Nous appliquons l'opérateur  $w()$  :

$$\underbrace{w[\Delta w(\Psi_i)]}_a = \Delta\Psi_i - \underbrace{2\pi(k_i - k_{i-1})}_b - 2\pi k \quad (\text{II. 10})$$

Où  $k_i, k_{i-1}$  représentent un multiple propre de  $2\pi$  alors :

$$w[\Delta w(\Psi_i)] = \Delta\Psi_i \quad (\text{II. 11})$$

Ce qui confirme l'équation II.3 et sa généralisation II.5.

### II.3 Déroulement par ajustement

Dans cette partie nous allons présenter ce qui est nécessaire d'une approche de déroulement de phase améliorée dite d'ajustement au niveau de franges [17] basée sur des études menées dans [18,19].

### II.3.1 traitement 1D

Le déroulement est l'attribution de nombre de cycle propre  $k(x_i)$

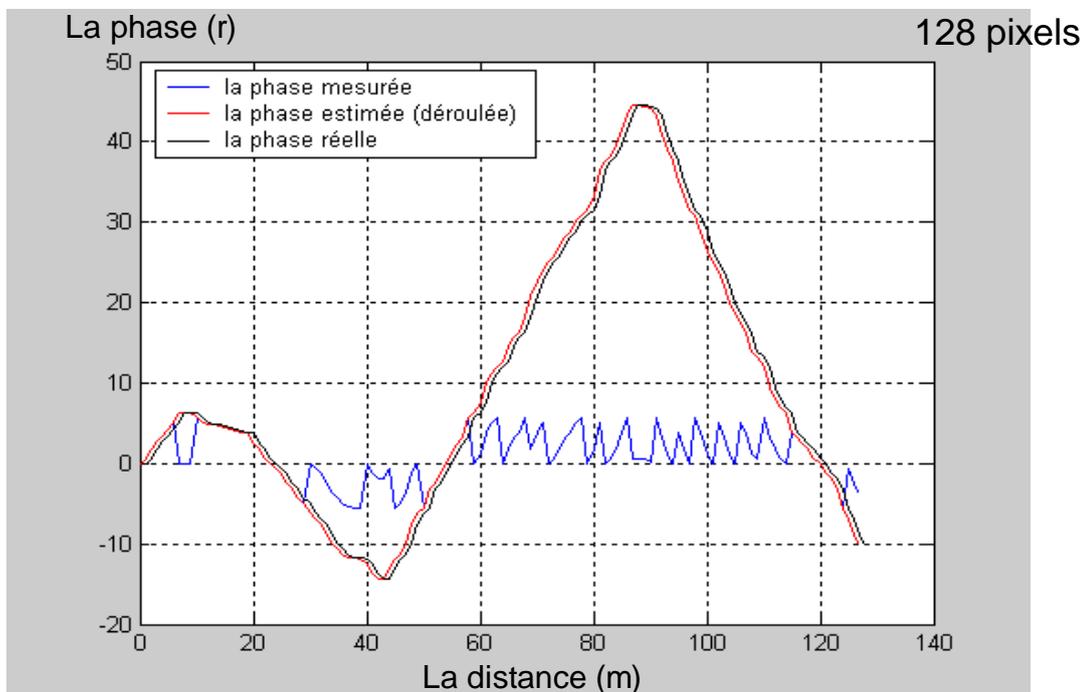
$$k(x_i) = \begin{cases} \sum_{\alpha=1}^i k_{\alpha}(x_{\alpha}); & 1 < i \leq N \\ 0; & i = 1 \end{cases} \quad (\text{II.12})$$

On définit la fonction  $k_i(x_i)$  pour chaque  $x_i \neq x_1$ , par :

$$k_i(x_i) = \begin{cases} 0 & , |\varphi_m(x_i) - \varphi_m(x_{i-1})| \leq \pi \\ 1 & , |\varphi_m(x_i) - \varphi_m(x_{i-1})| > \pi \text{ et } \varphi_m(x_{i-1}) > \varphi_m(x_i) \\ -1 & , |\varphi_m(x_i) - \varphi_m(x_{i-1})| > \pi \text{ et } \varphi_m(x_{i-1}) < \varphi_m(x_i) \end{cases} \quad (\text{II.13})$$

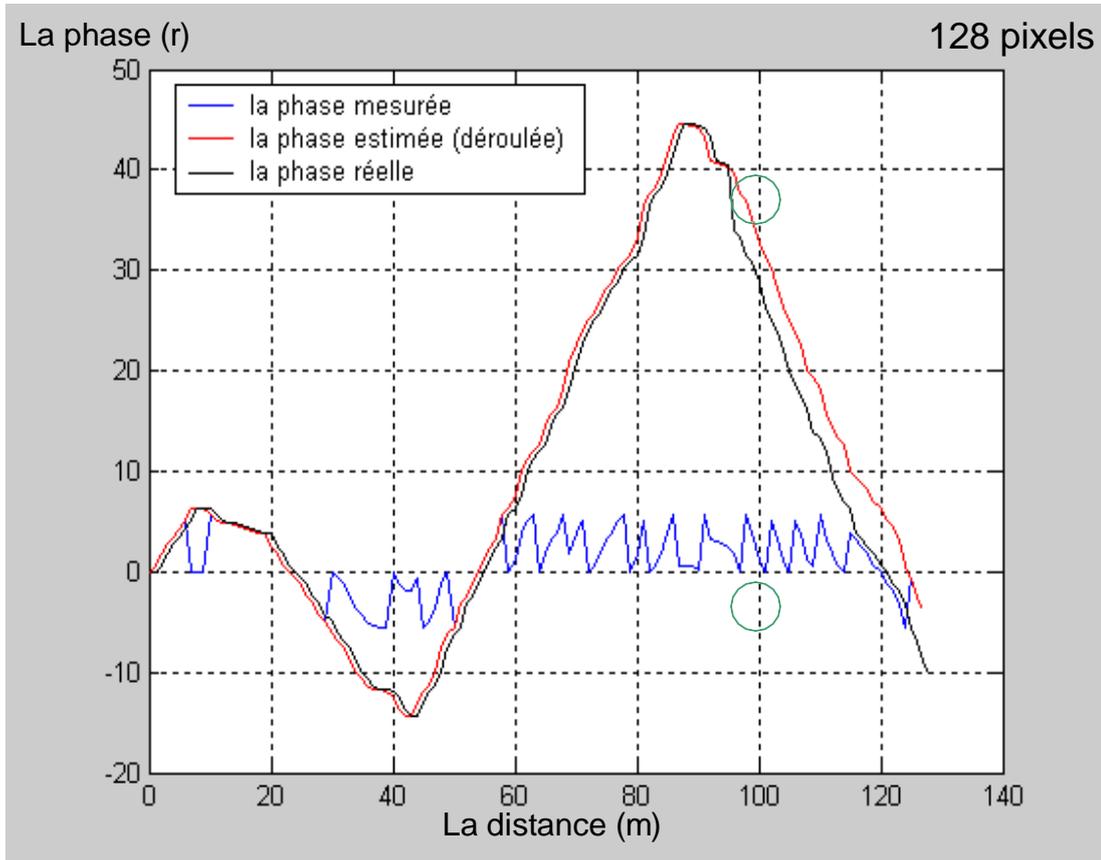
Alors chaque pixel  $x_i$  est attribué un nombre de cycle propre (appelé aussi ordre) qui représente un ajustement avec le pixel adjacent  $x_{i+1}$  et les nombres de cycles de tous les pixels qui lui précèdent.

La figure ci-dessous montre un exemple d'un traitement 1DA qui représente une section d'interférogramme pour un vecteur de 128 pixels :



**Figure. II.2** : Traitement unidimensionnel pour une section de la surface, condition d'Itoh satisfaite

D'après la figure II.2 qui représente une ligne ou colonne d'image, nous remarquons que la courbe de la phase déroulée est proche de celle réelle parce que toutes les variations entre les pixels sont inférieures ou égales à  $\pi$ , la principale condition de faisabilité de déroulement de phase est satisfaite. Par contre celles de la figure II.3 ne sont pas identiques où un décalage se crée à partir de pixel encerclé parce que la variation de phase au niveau de ce pixel dépasse  $\pi$ .



**Figure. II.3 :** Traitement unidimensionnel pour une section de la surface, violation de la condition d'Itoh

### II.3.2 traitement 2D

La matrice d'attribution des nombres de cycle est :

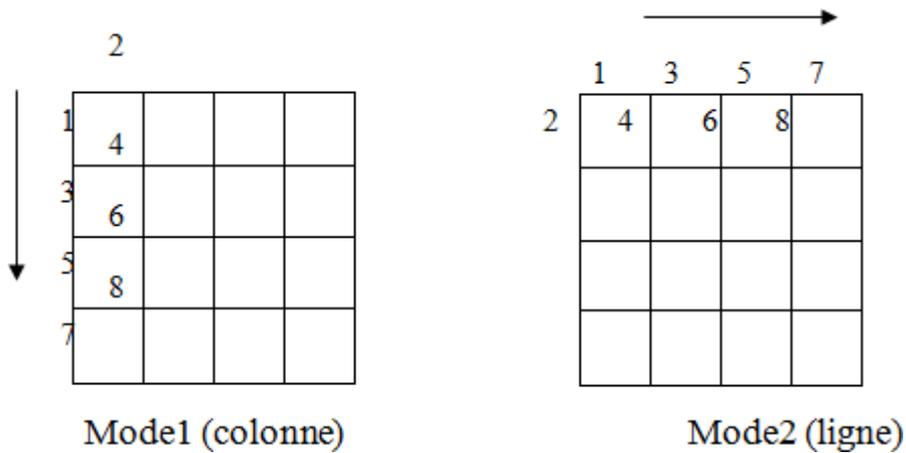
$$k(x_i, y_j) = \begin{cases} \sum_{\alpha=1}^i k_{\alpha}^c(x_{\alpha}, y_j) + \sum_{\beta=1}^j k_{\beta}^l(x_i, y_{\beta}); & 1 < i \leq N \text{ et } 1 < j \leq M \\ 0; & i = 1 \text{ et } j = 1 \end{cases} \quad (\text{II.14})$$

Où :

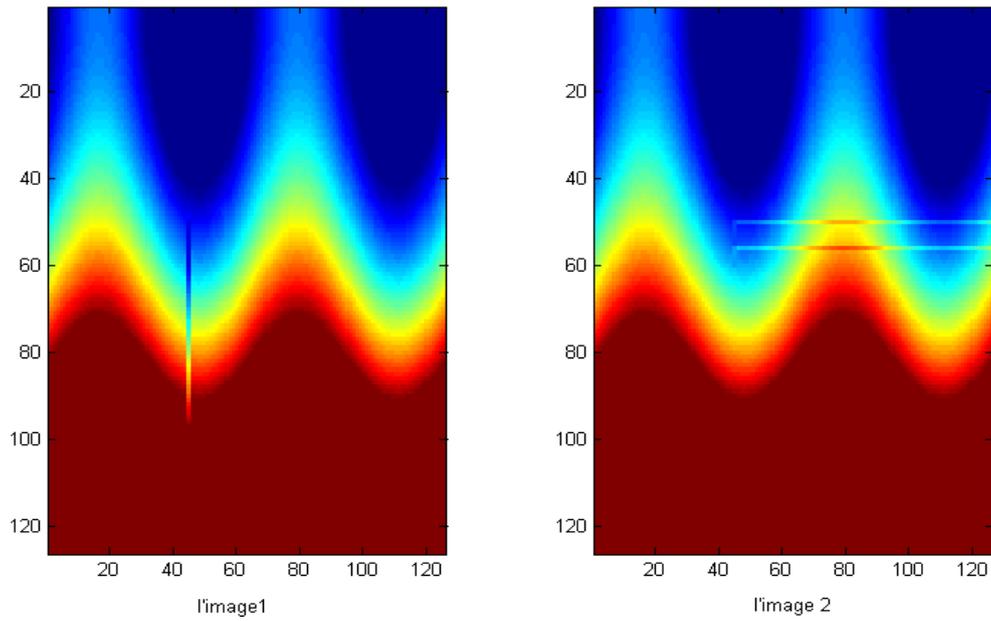
$$k_i^c(x_i, y_j) = \begin{cases} 0 & , |\varphi_m(x_i, y_j) - \varphi_m(x_{i-1}, y_j)| \leq \pi \\ 1 & , |\varphi_m(x_i, y_j) - \varphi_m(x_{i-1}, y_j)| > \pi \text{ et } \varphi_m(x_{i-1}, y_j) > \varphi_m(x_i, y_j) \\ -1 & , |\varphi_m(x_i, y_j) - \varphi_m(x_{i-1}, y_j)| > \pi \text{ et } \varphi_m(x_{i-1}, y_j) < \varphi_m(x_i, y_j) \end{cases}$$

$$k_j^l(x_i, y_j) = \begin{cases} 0 & , |\varphi_m(x_i, y_j) - \varphi_m(x_i, y_{j-1})| \leq \pi \\ 1 & , |\varphi_m(x_i, y_j) - \varphi_m(x_i, y_{j-1})| > \pi \text{ et } \varphi_m(x_i, y_{j-1}) > \varphi_m(x_i, y_j) \\ -1 & , |\varphi_m(x_i, y_j) - \varphi_m(x_i, y_{j-1})| > \pi \text{ et } \varphi_m(x_i, y_{j-1}) < \varphi_m(x_i, y_j) \end{cases}$$

Cette description mathématique nous donne une explication plus détaillée que les équations II.3,4 et 5, et son application donne une sorte d'ajustement inter-pixels. Selon ces ajustements deux modes de parcours simples sont possibles. Les figures II.4, 5 et 6 ci-dessous montrent ce propos.

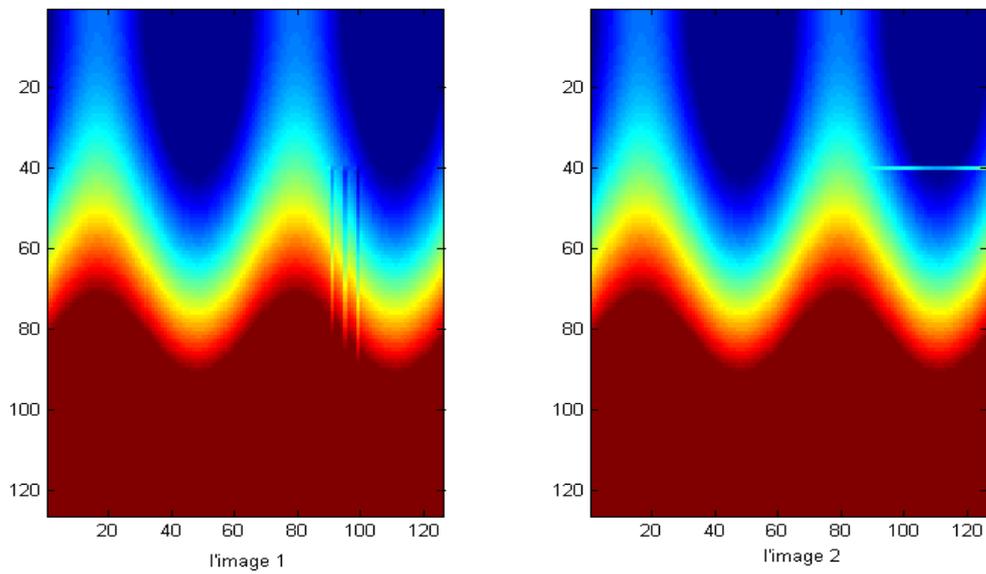


**Figure II.4 :** Les deux modes simples de la méthode d'ajustement



**Figure II.5 :** Les deux images générées pour des sauts de phase situés entre

$$x_{50} \leq x_i \leq x_{60} \quad y_{40} \leq y_j \leq y_{45}$$

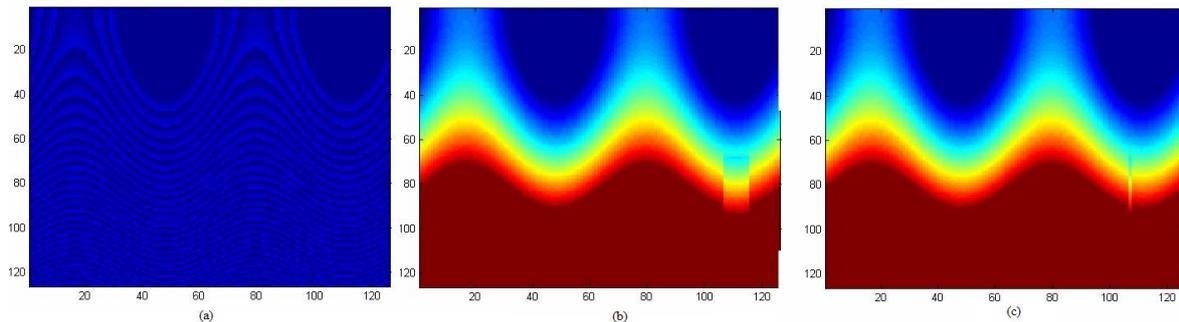


**Figure II.6 :** Les deux images générées pour des sauts de phase situés entre

$$x_{35} \leq x_i \leq x_{40} \quad y_{90} \leq y_j \leq y_{100}$$

Pour un intréférogramme sans bruits l'application de la méthode basique de gradient (l'équation II.5) ou la méthode d'ajustement (l'équation II.14) sont similaires et donnent des

images de phases déroulées proches. Mais avec bruits les lignes distordues avec la méthode d'ajustement sont légèrement moins larges.



**Figure II.7** : Le déroulement avec résidus et les lignes distordues. (a) l'interférogramme, (b) la méthode de gradient basique, (c) la méthode d'ajustement.

## II.4 autres approches de déroulement

Il y a deux catégories pour le déroulement : suivi de parcours (*path-following*) et les normes de minimisation (*Minimum-Norm*). Dans cette partie nous essayerons d'expliquer les algorithmes les plus connus et répandus, cet état de l'art permettra de simplifier les différentes procédures et se focaliser directement sur l'application sans oublier de citer les avantages et les inconvénients de chaque catégorie et algorithme afin de donner une analyse assez complète que possible.

### II.4.1 Les méthodes de *Branch-cut*

Ces méthodes appartiennent à la catégorie de suivi de parcours (*Path-following*). Elle était proposée pour la première fois par *Goldstein* [20], elle est considérée parmi les algorithmes de référence pour la comparaison et la validation d'autres méthodes améliorées. Son principe est basé sur l'établissement des branches entre les résidus de signes opposés dans le but d'interdire l'intégration traversant ces branches d'où vient le nom *branch-Cut* [20-22].

#### II.4.1.1 La méthode de Goldstein

L'exécution de l'algorithme se déroule comme suit :

1. Générer la mappe des résidus.
2. Connecter les résidus de différents signes suivant la distance la plus courte par des branches qui servent comme des barrières interdisant le parcours d'intégration de les traverser, de telle sorte qu'aucun résidu net ne peut être encerclé et aucune propagation d'erreur ne peut être engendrée. Mais certaines erreurs locales (régions distordues sans

déroulement) peuvent se produire. Bien sûr les pixels des extrémités de chaque branche sont de signe opposé donc cette branche représente la discontinuité de phase. Le but est de choisir les branches de telle manière de minimiser la longueur totale des branches et par conséquent minimiser les discontinuités et les régions restées sans déroulement.

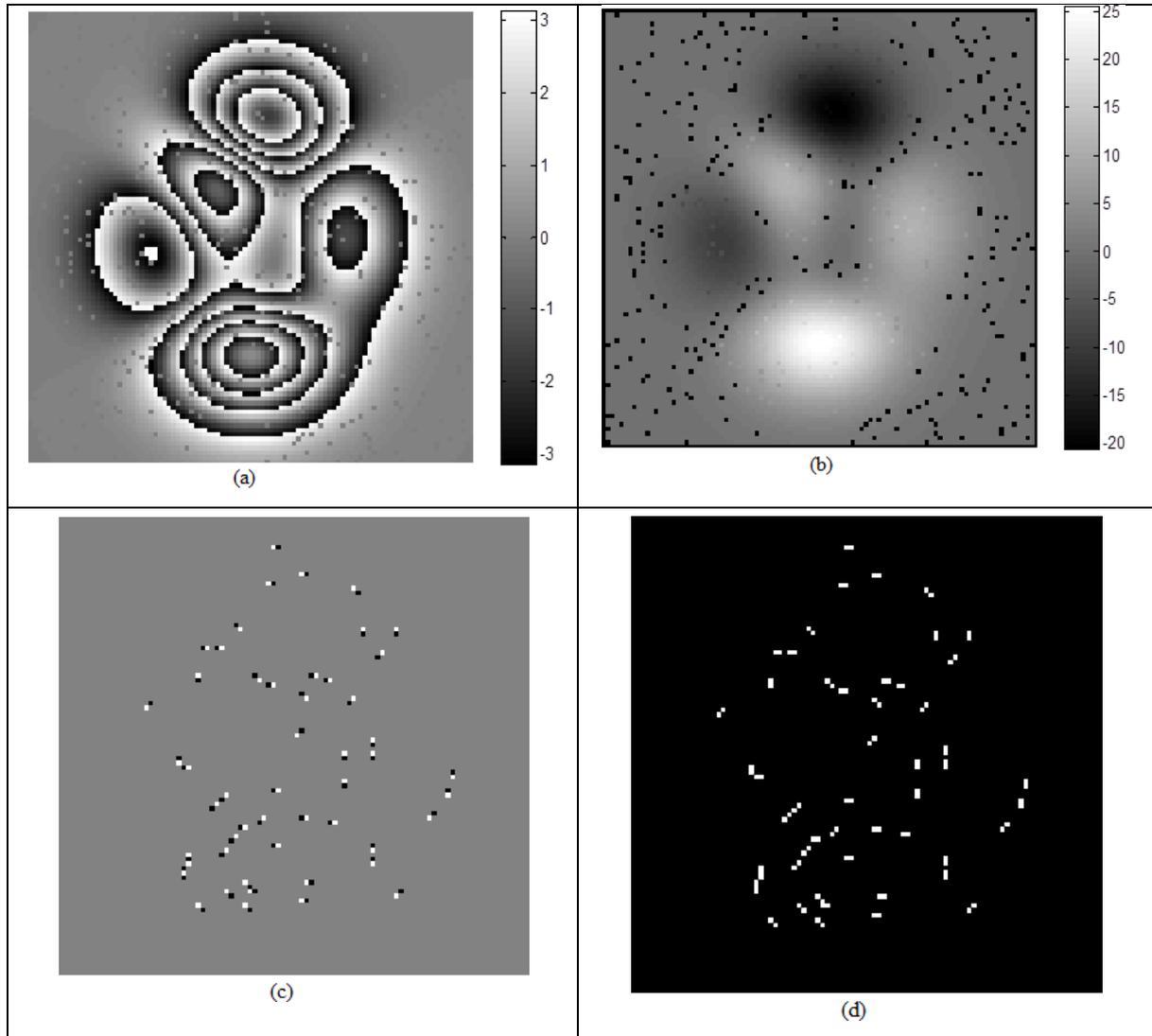
Quand le taux des résidus est faible la localisation des branches optimums est très simple. Mais quand la densité des résidus est élevée la performance de sélectionner les bonnes branches se diminue. Alors pour les régions les plus bruitées il apparaît que le seul moyen de minimiser la longueur totale des branches est d'essayer toutes les combinaisons  $n(n-1)/2$  ce qui induit à un temps de traitement si lourd.

*Goldstein* a proposé l'algorithme de connexion des résidus tel que :

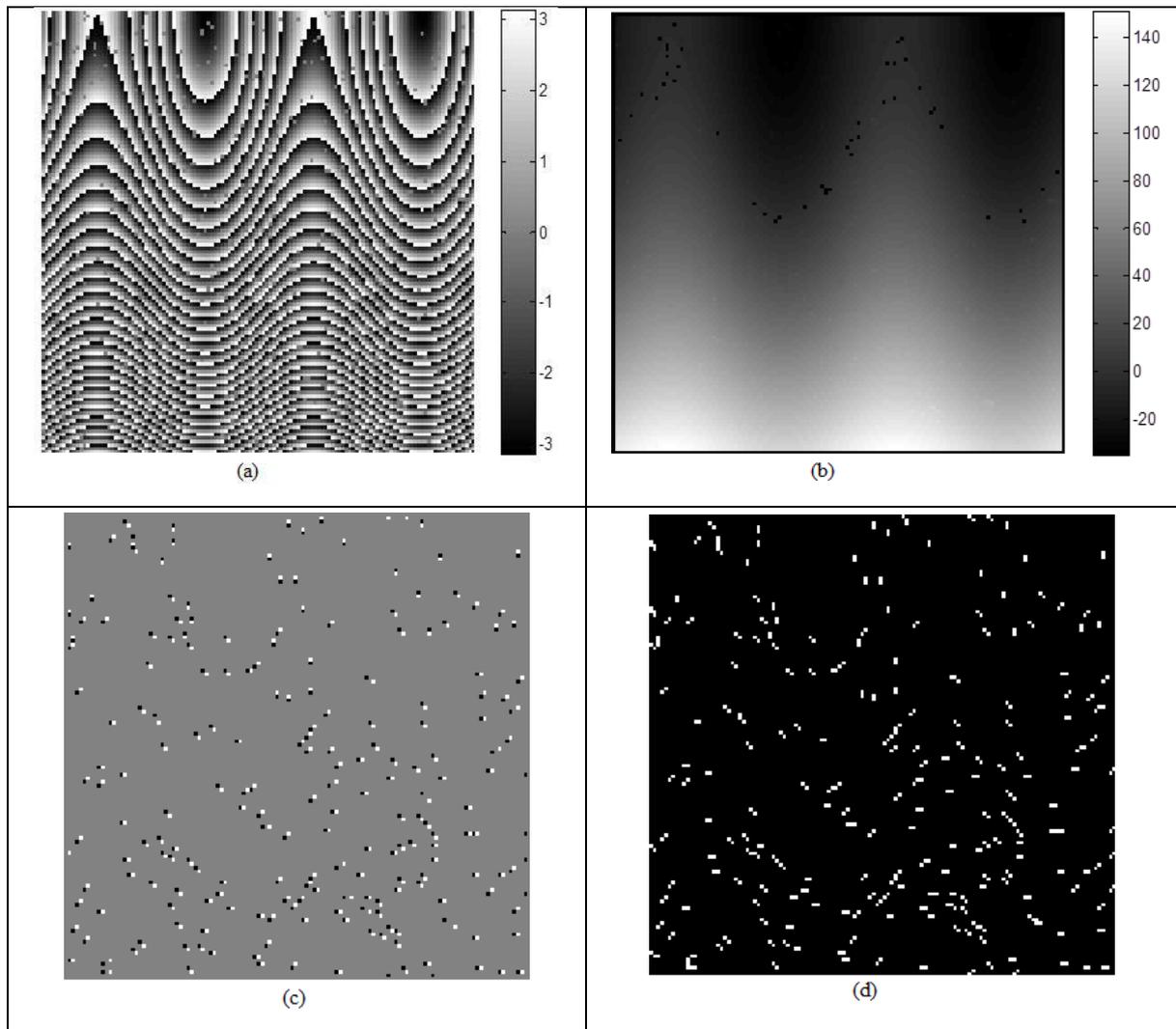
- a) Un masque 3 x 3 scanne les voisins de chaque résidu. Dans ce masque les résidus voisins sont inspectés et connectés aux résidus central en formant une branche, la branche formée se marque soit balancée (non chargée) ou non balancée (signe +/-)
  - b) Incrémentation de masque par 2 et le même algorithme se répète à partir de résidu de départ. On s'arrête s'il y a aucun résidu hors les grappes de branches et toutes les branches sont balancées.
  - c) S'il y a des résidus restant ou des grappes non balancées nous pouvons les connecter au plus proche bord.
  - d) Quand les résidus sont très denses toute la région sera isolée et marquée balancée.
3. Finalement l'intégration des gradients s'exécute en évitant toutes les branches et les régions isolées, cette étape est appelée (*Flood-Fill*) c.-à-d. un remplissage simulé par un écoulement inondant.

Les figures ci-dessous montrent ce propos, l'algorithme a été testé sur deux interférogrammes simulés (figures II.8-9.a) et un autre réel constitue une partie d'interférogramme fournie par d'ERS-1 (figure II.10.a). Le premier présente un faible taux de franges fermées, le deuxième un grand taux de franges ouvertes et le troisième présente une donnée inSAR réelle de grand taux de résidus. Les figures II.8-10.b montrent les images de phase déroulée par l'algorithme de *Branch-Cut*, les figures II.8-10.c et II.8-10.d présentent respectivement les mappes de résidus et celles de branches correspondants.

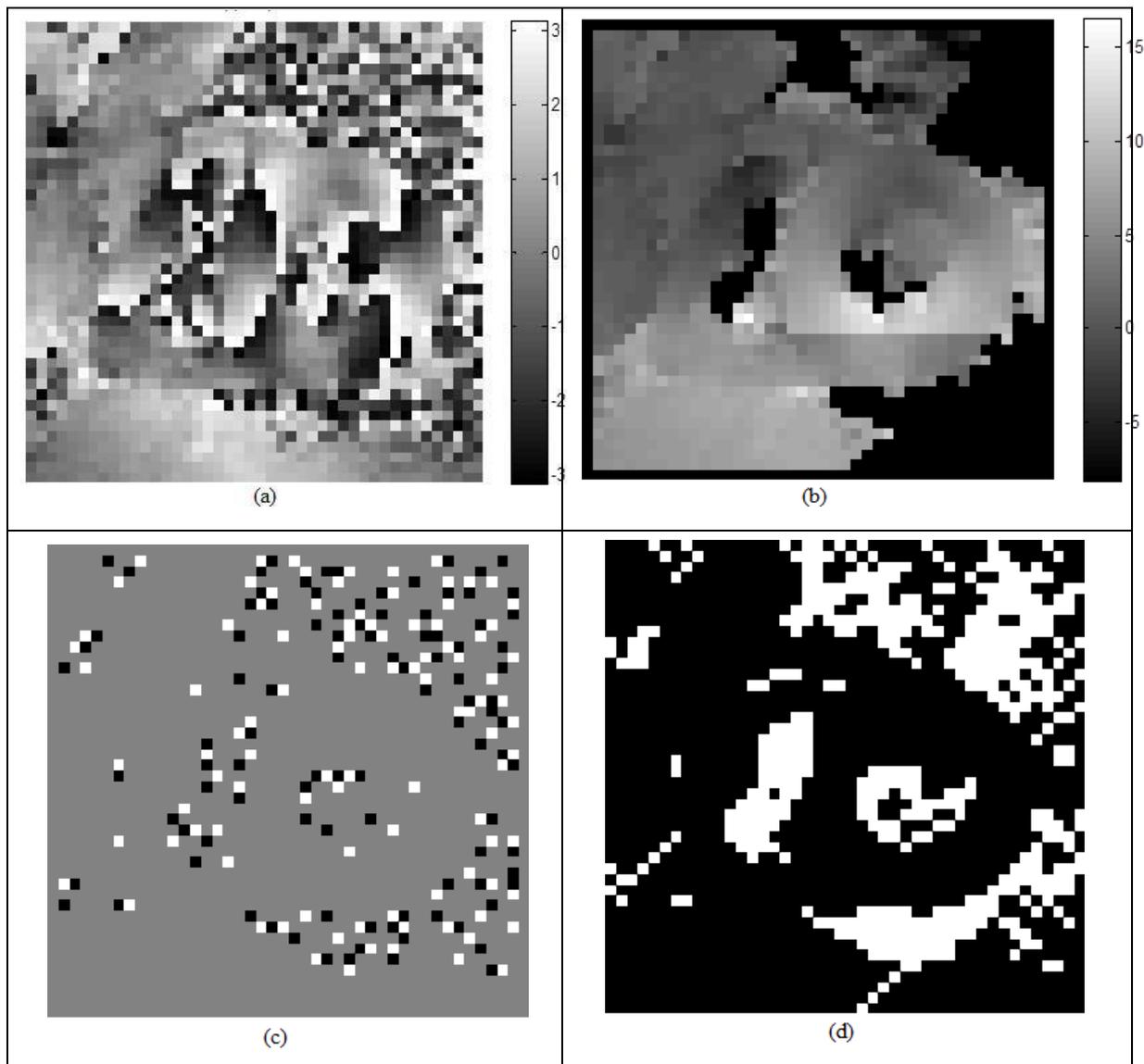
Nous remarquons que les images déroulées contiennent des régions isolées où aucune procédure d'intégration de gradient n'est effectuée, ces régions sont de densité élevée de résidus. Donc la clé d'amélioration de la méthode de *Branch-Cut* est de minimiser ces régions.



**Figure II.8** : Test d'algorithme de déroulement Goldstein pour un interférogramme pics à franges fermées. (a) l'interférogramme, (b) l'image de phase déroulée, (c) la mappe des résidus, (d) la mappe de branches



**Figure II.9** : Test d'algorithme de déroulement Goldstein pour un interférogamme ondes à franges ouvertes.(a) l'interférogamme, (b) l'image de phase déroulée, (c) la mappe des résidus, (d) la mappe de branches



**Figure II.10** : Test d'algorithme de déroulement Goldstein pour une partie d'un interférogramme réel d'ERS-1. (a) l'interférogramme, (b) l'image de phase déroulée, (c) la mappe des résidus, (d) la mappe de branches

L'algorithme de *Branch-Cut* de *Goldstein* est rapide et généralement satisfaisant mais ses lacunes sont :

- Région isolées dans les grappes formées par l'agglomération des branches.
- La procédure de connexion des résidus (créations des branches) se fait pixel par pixel et aucun processus de recherche globale n'est appliqué pour former des grappes optimums dont la longueur totale des branches est minimisée.

Autres améliorations de l'algorithme ont été proposées, elles incluent une étape d'optimisation ; soit après la formation des grappes soit après celle des branches ; permettant d'inter-changer

les résidus entre les grappes afin de réduire la longueur totale des connexions. Parmi ces méthodes nous en trouvons [23-25].

#### II.4.1.2 Méthode *Branch-Cut* guidée par la fonction pseudo corrélation

Dans la littérature [26] il s'est proposé une nouvelle méthode améliorant le processus de placement des branches afin de minimiser la longueur totale et de relier les discontinuités de phase effectives.

La mappe de qualité utilisée est la pseudo-corrélation de la phase roulée, elle est définie telle que :

$$PC(m, n) = \frac{\sqrt{(\sum \cos \phi(i, j))^2 + (\sum \sin \phi(i, j))^2}}{k^2} \quad (\text{II. 15})$$

Où la valeur  $PC(m, n)$  juge la qualité de la phase  $\phi(m, n)$  avec les voisins  $k \times k$ ,

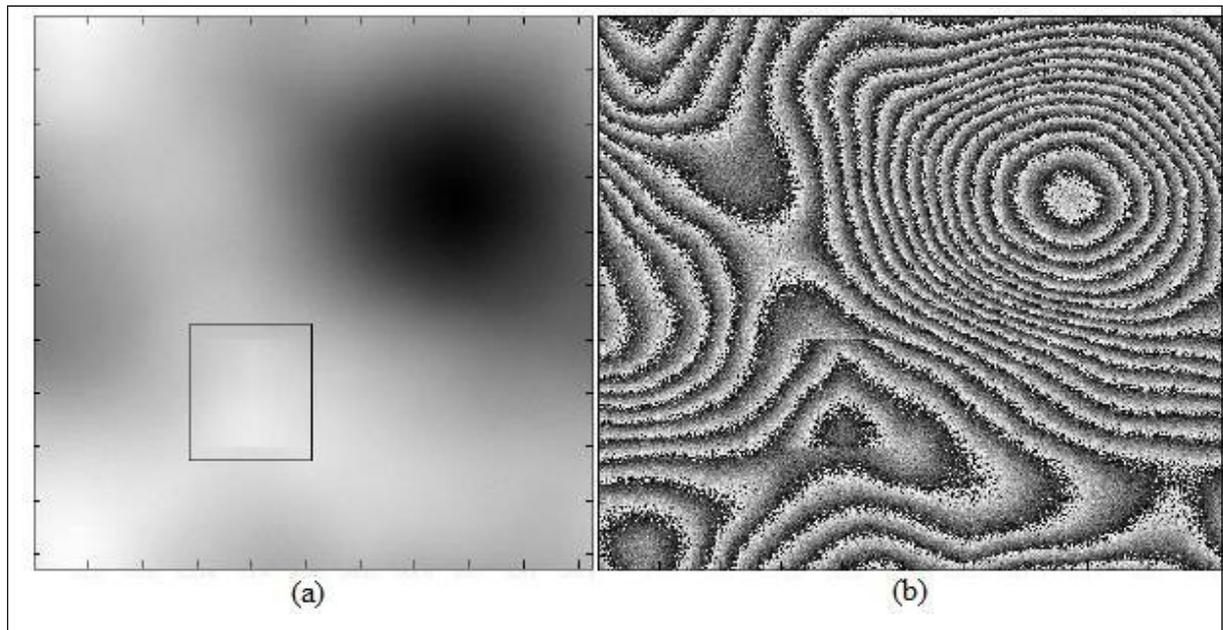
L'algorithme se déroule comme suit :

- 1) Générer la mappe des résidus.
- 2) Définir un seuil la longueur max de branche placée dans cette étape. Les résidus de signe inverse séparés par une distance inférieure au seuil seront marqués. Ces résidus marqués seront balancés et placés comme une branche. Puis nous les élevons de la mappe des résidus. Le choix de seuil est empirique et selon l'estimation de manipulateur par exemple dans [26] le seuil 3 a été choisi.
- 3) Générer la mappe de pseudo-corrélation décrite dans l'équation II.15. Puis générer la mappe de guide par la procédure de détection des bords [27] à partir de la mappe de pseudo-corrélation.
- 4) Placer les résidus qui se trouvent sur un bord de la mappe de guide en cherchant les résidus de signe inverse le long du bord. Si aucun résidu inverse n'est trouvé sur le bord, rechercher un résidu inverse autour de chaque extrémité de bord. Arrêter de chercher dans cette direction lorsque nous rencontrons un résidu de même signe.
- 5) Pour les résidus restant non-balancés placer les branches selon l'algorithme de *Goldstein*. Et enfin fusionner toutes les branches.

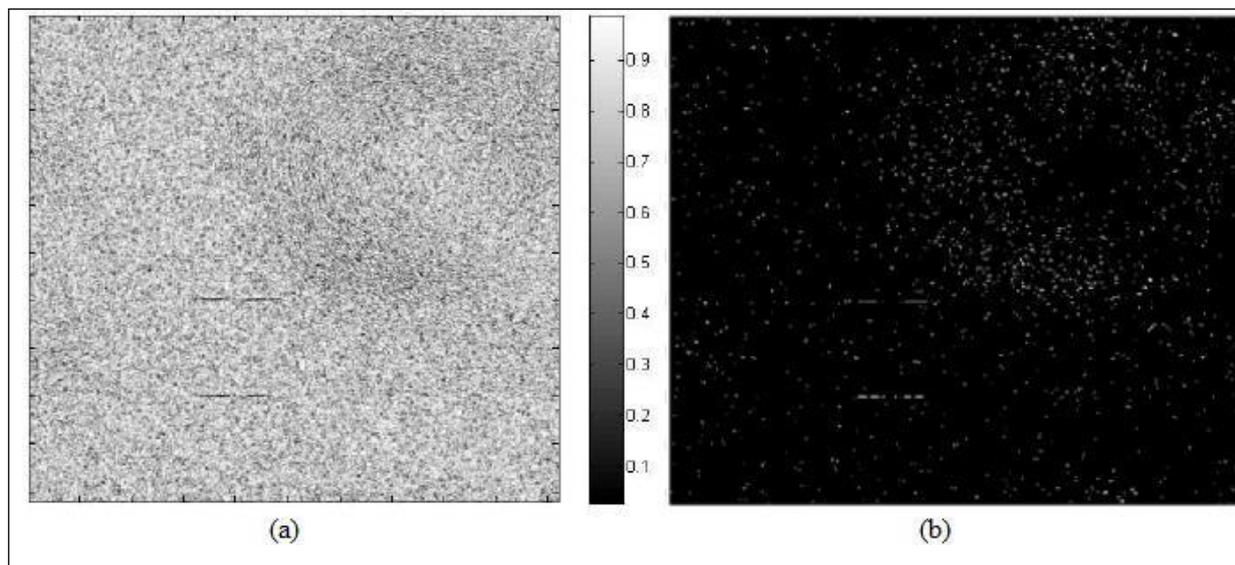
L'efficacité de cet algorithme se manifeste et apparait différente de celle de Goldstein lorsque nous avons des discontinués raides dans l'interférogramme. Dans l'exemple de la figure ci-dessous nous avons une image de phase originale avec des discontinués raides artefacts sous forme d'un carré (figure II.11. a), son interférogamme correspondant est illustré dans la figure

II.11.b. La détection de ces discontinuités est visible sur la figure II.12.a-b en utilisant la mappe de pseudo-corrélation et la celle de guide déduite par la procédure de détection des bords.

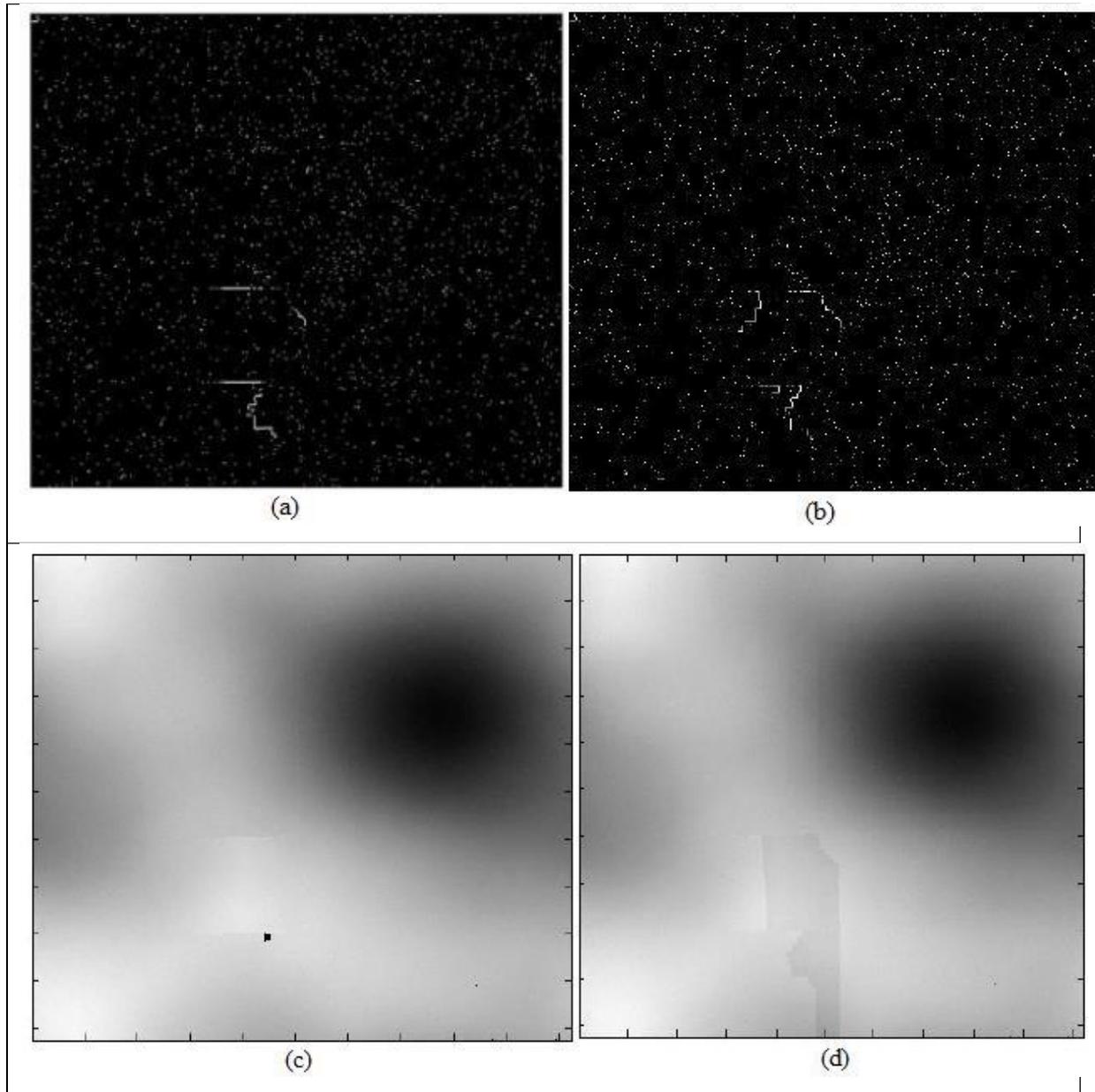
La différence entre l'algorithme de Goldstein de celui guidée par la pseudo-corrélation se réside au niveau de longueur des branches et leur densité dans l'image (figures II.13.a-b) et par conséquent minimiser les régions distordues lors de déroulement de phase (figures II.13.c-d).



**Figure II.11** : (a) l'image de phase originale avec discontinuités raides, (b) l'interférogramme correspondant [26]



**Figure II.12** : (a) la mappe de pseudo-corrélation, (b) la mappe de guide obtenue par la détection des bords.



**Figure II.13 :** (a) la mappe des branches guidée par la pseudo-corrélation, (b) la mappes des branches de Goldstein, (c) l'image de phase déroulée après le placement de branches par GPC, (d) l'image de phase déroulée après le placement de branches de Goldstein.

#### II.4.1.3 Méthode *Branch-Cut* d'appariement par distance

Dans [28] une autre méthode placement des branches a été proposée, cet algorithme est basé sur l'agrégation des coordonnées pour chaque résidu positif (ou négatif) toujours dans le but de trouver un résidu inverse le plus proche tout en cherchant à avoir une longueur totale minimum

que possible. La méthode d'appariement par distance (AD) place ses branches selon une vision globale de tous les résidus pour éviter de créer des zones isolées par des régions non-chargées massives. Ainsi tous les résidus restant non balancés peuvent être connectés aux bords de l'interférogramme.

L'algorithme AD se déroule comme suit :

- 1) Générer la mappe des résidus, déterminer séparément les coordonnées des résidus positifs et négatifs (une sorte de deux matrices séparées et superposées).
- 2) Calculer les distances entre les résidus avec restriction c.-à-d. pas  $n(n-1)/2$  possibilités.

Le paramètre distance (noté ici *dis*) se calcule par :

3)

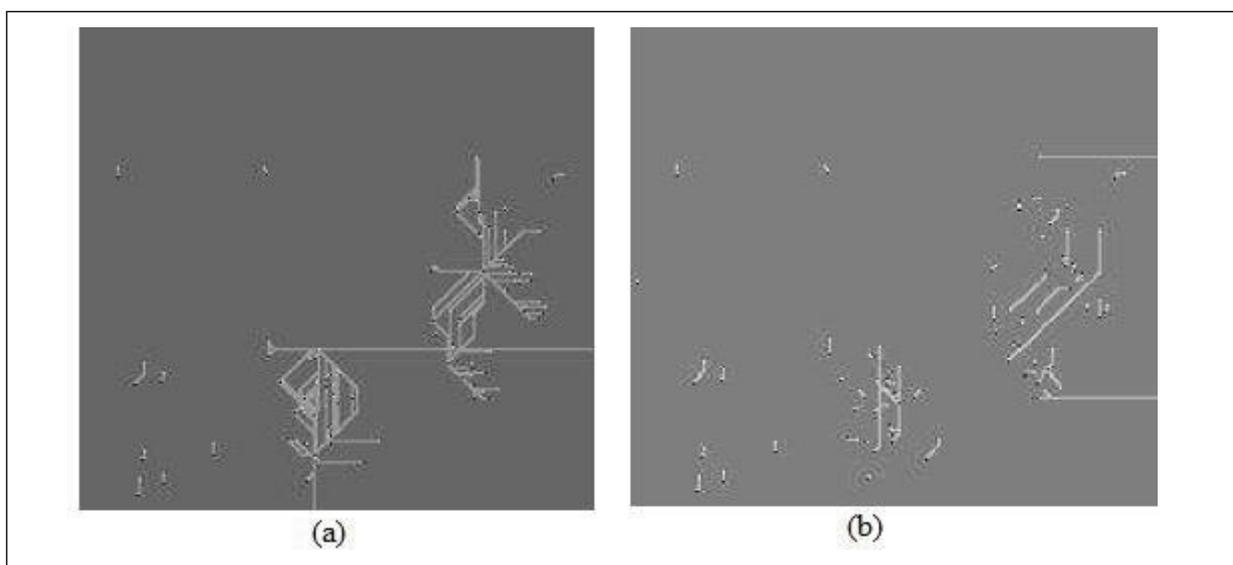
$$dis = \sqrt{(i_- - i_+)^2 + (j_- - j_+)^2} \quad (\text{II. 16})$$

Ou; pour des raisons d'éviter les fractions décimales, par :

$$dis = \max(|i_- - i_+|, |j_- - j_+|) \quad (\text{II. 17})$$

- 4) Initialiser le paramètre *dis* à 1 et connecter les résidus satisfaisants cette distance.
- 5) Incrémenter *dis* et répéter la procédure.
- 6) Connecter les résidus restés non balancés au bord.

La figure II.14 montre la différence entre le placement des branches par la méthode de Goldstein et AD. Nous remarquons que la méthode de Goldstein forment des branches agglomérées et plus massive avec des plus des régions isolées.



**Figure II.14** : Comparaison entre la mappe des branches de Goldstein et AD.

(a) Goldstein, (b) AD [28]

## II.4.2 Les méthodes de qualité guidée et la croissance de la région

Le déroulement de phase par la qualité guidée est une approche appartient à la catégorie (*Path-Following*), elle est essentiellement basée sur la génération d'une mappe de qualité d'interférogramme à partir de celle de cohérence originale (amplitude de corrélation) issue conjointement de processus d'imagerie inSAR ou à partir d'une autre déduite de l'interférogramme telle que la variance ou la pseudo corrélation dans le cas où l'image de cohérence n'est pas disponible [29-32].

Le processus d'obtention de la mappe de qualité est communément appelé (*region-growing*) en le simulant à une croissance d'une région cultivée. L'image de qualité peut être utilisée comme une mappe guidant le filtrage d'interférogramme, ou une mappe servant à guider le déroulement de phase et dans ce cas nous nous trouvons face à deux possibilités d'utilisation :

- Une mappe de qualité définissant les grappes et les régions isolées résultant des branches qui contiennent des pixels de mauvaise qualité à n'est pas dérouler [33].
- Une mappe de qualité définissant les régions saines et consistantes qui contiennent des pixels de bonne qualité à dérouler [30, 32, 34, 35].

Autrement dit le processus de (*region-growing*) consiste tout simplement à diviser l'interférogramme en deux zones : bonne et mauvaise selon un ou plusieurs seuils.

### II.4.2.1 La méthode de Flynn

*Flynn* a proposé pour la première fois d'utiliser une mappe de qualité [33] pour couvrir les régions de grappe non-chargées (équivalentes aux branches de *Goldstein*) par un masque, le masque qui couvre ces régions de mauvaise qualité devrait être de petite surface que possible pour garder les détails d'information de phase dans l'interférogramme et cela faisable uniquement si le taux des résidus n'est pas trop grand.

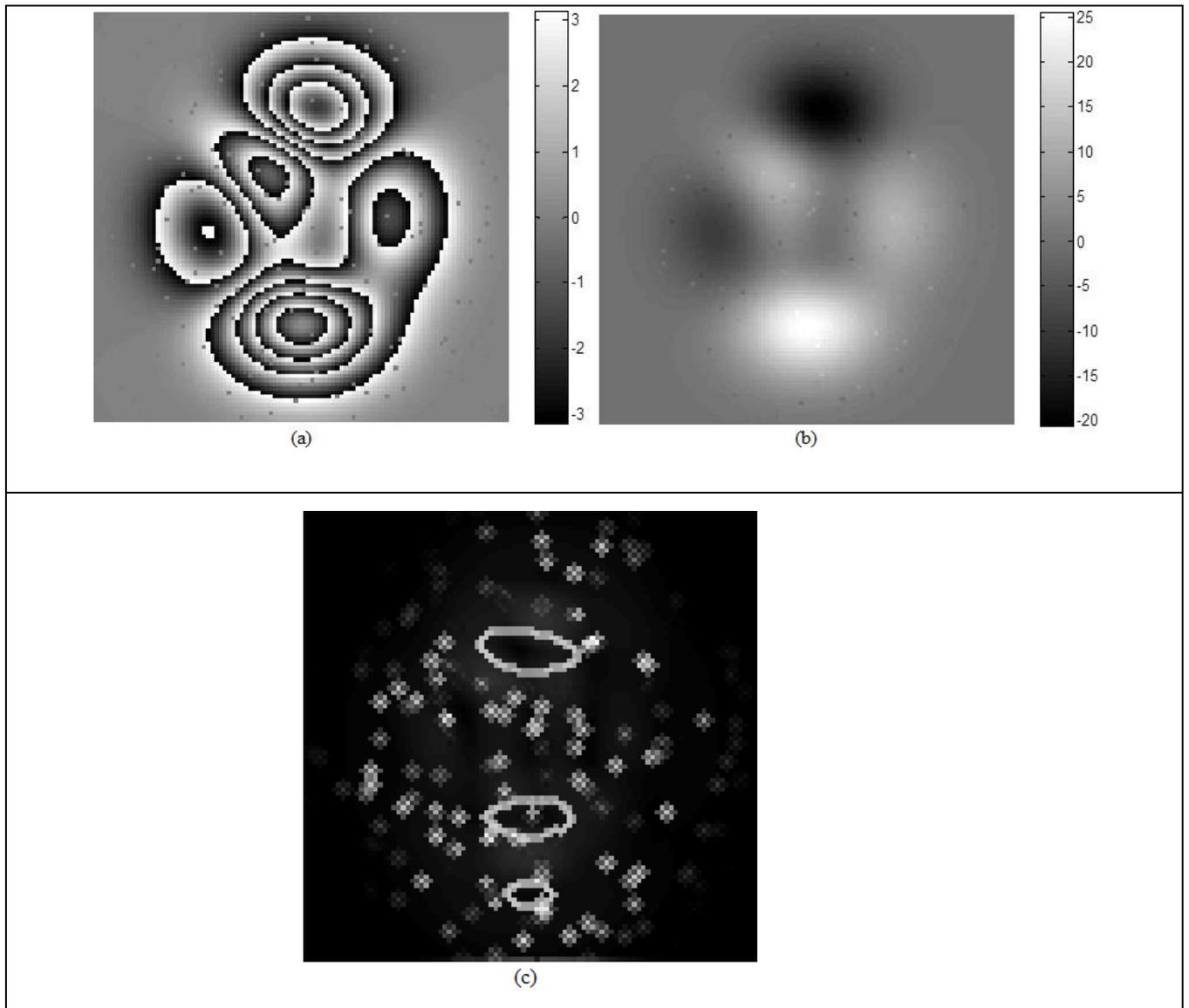
Cette méthode se déroule avec conception contraire à celle de *Roth* et ses variantes améliorées, au lieu de faire croître la zone de déroulement à travers les pixels de bonne qualité elle fait croître le masque à travers les mauvais pixels en commençant à partir des résidus. Elle poursuit les résidus de chaque composante connectée au masque (le masque comporte plusieurs composantes et chaque composante doit comporter des résidus de signe balancé). Elle termine quand toutes les composantes soient balancées. Puis le déroulement de phase s'exécute par le processus (*Flood-Fill*) en évitant tous les pixels de masque.

L'algorithme se déroule comme suit :

- 1- Générer la mappe des résidus et celle de qualité d'interférogramme (cohérence, pseudo-corrélation ou la variance) avec sa fenêtre minimum 3x3.
- 2- Tous les pixels résidus et ses voisins qui l'entourent se déclarent initialement comme graines pour le processus de (*Region-Growing*) et se mettent dans le masque initial.
- 3- Chercher parmi les voisins des pixels graines (sauf les résidus) ceux qui ont une qualité inférieure à un seuil prédéterminé pour les ajouter au masque.
- 4- Le seuil sera incrémenté quand la croissance devient impossible. Comment se trouver dans cette situation ? si aucun pixel n'est ajouté et certains résidus ne sont pas dans les composantes balancées (restent dans la zone de déroulement).  
À noter que le seuil initial et la fenêtre de la mappe de qualité sont les seuls paramètres ajustables dans l'algorithme.
- 5- Le processus se termine quand toutes les composantes de masque soient balancées et aucun résidu ne se trouve hors le masque.

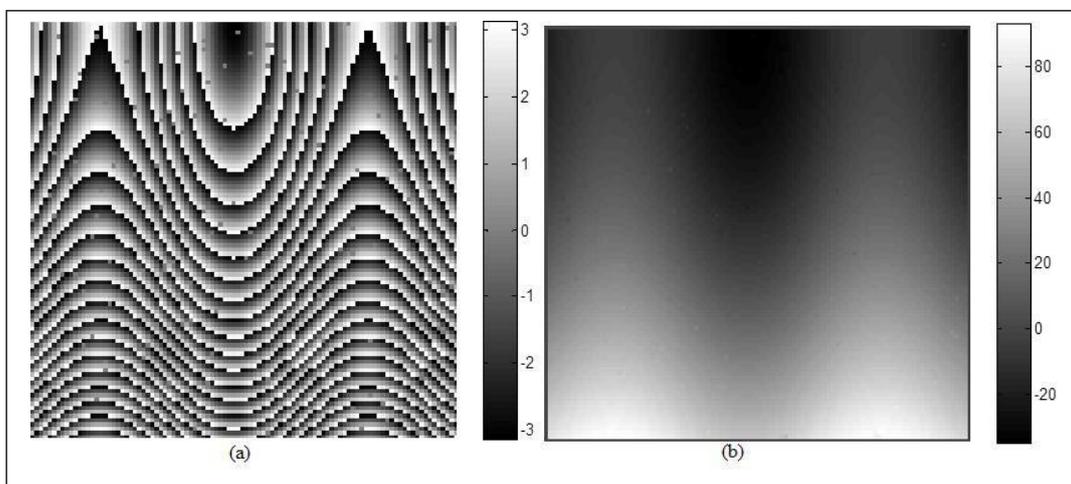
Pour le test les mêmes interférogrammes utilisés dans l'algorithme de *Goldsetin* ont été pris, il s'agit d'un interférogramme simulé aux franges fermées (figure II.15.a), un autre aux franges ouvertes (figure II.16.a) et une partie d'interférogramme réel ERS-1 (figure II.17.a). Les figures II.15-17. b présentent respectivement les images de phase déroulée par la méthode de *Flynn* pour chaque interférogramme. Tandis que les figures II.15-17.c montrent les mappes de qualité correspondante, la fonction de qualité que nous avons utilisé pour ce test est la variance dérivative de la phase (PDV)

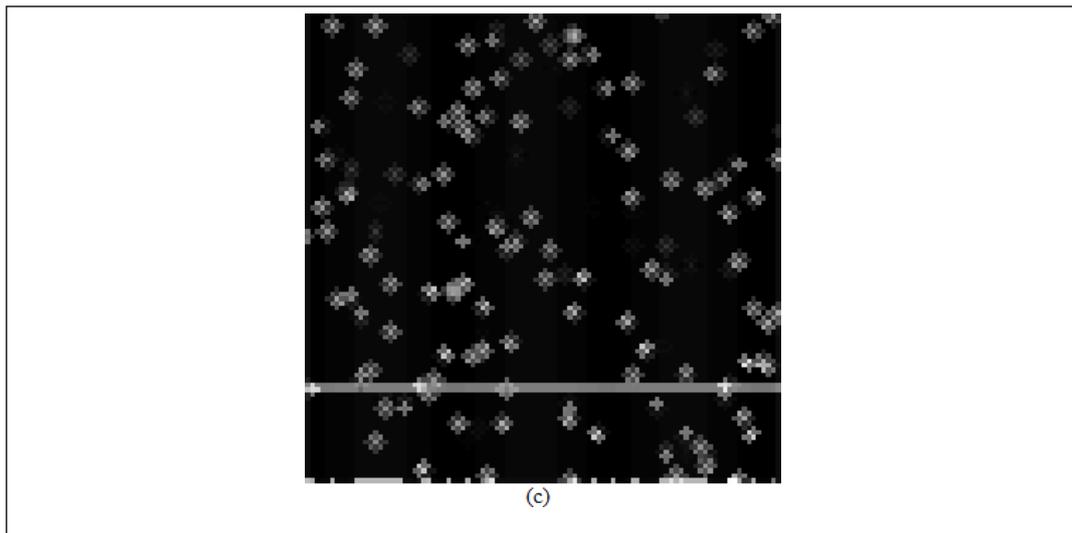
Nous remarquons que l'algorithme de *Flynn* a pu dérouler les zones qui auraient été isolées par l'algorithme de *Goldsetin*.



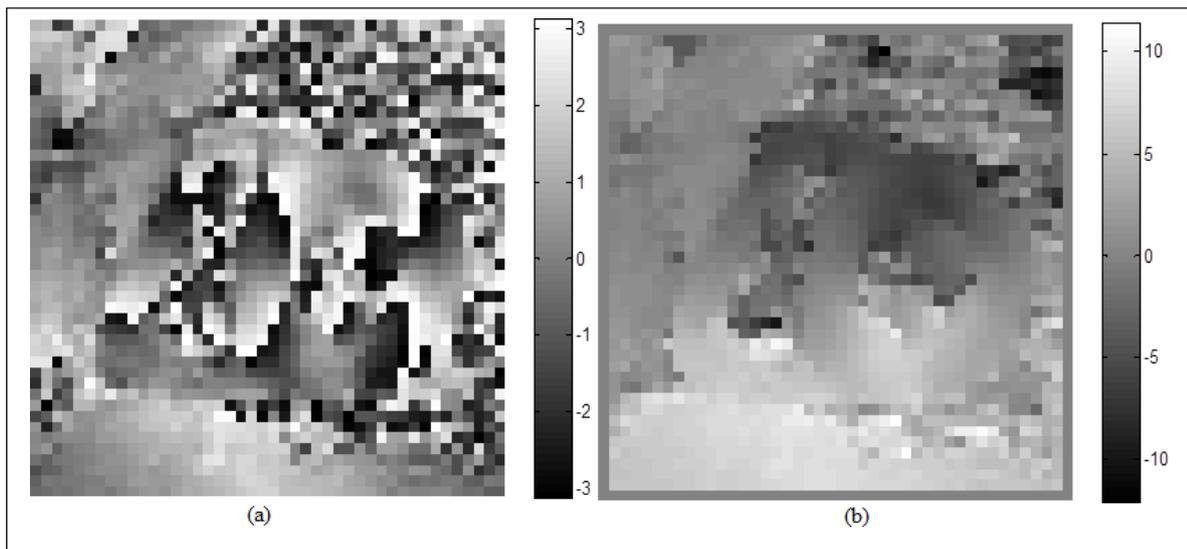
**Figure II.15** : Test d'algorithme de déroulement de Flynn pour un interférogamme pics à franges fermées. (a) l'interférogamme, (b) l'image de phase déroulée, (c) la mappe de qualité

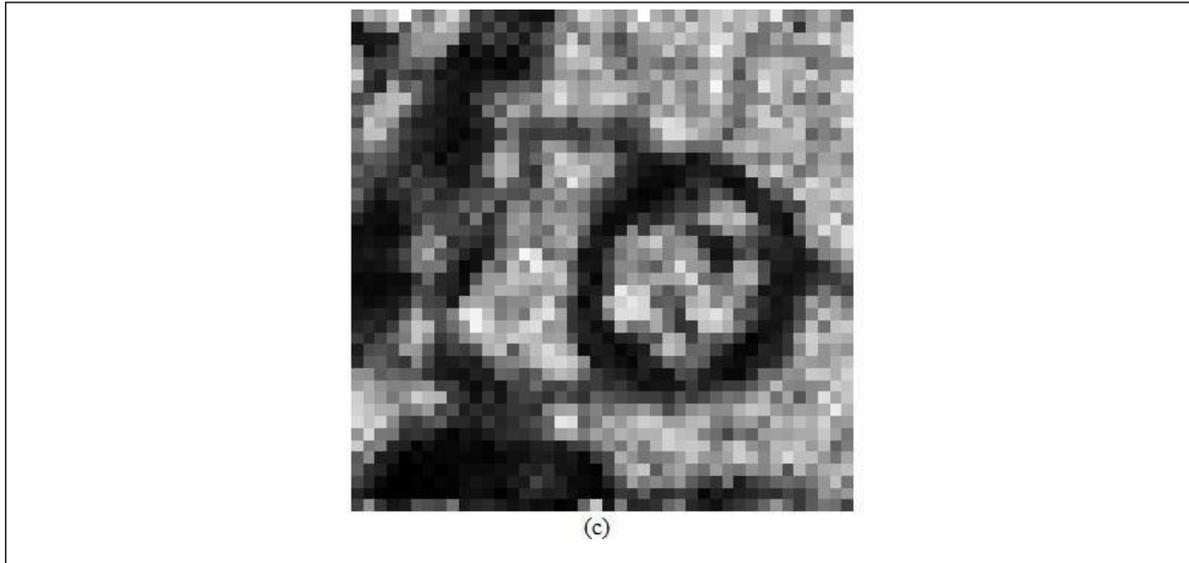
PDV





**Figure II.16** : Test d'algorithme de déroulement de Flynn pour un interférogramme ondes à frange ouverte. (a) l'interférogramme, (b) l'image de phase déroulée, (c) la mappe de qualité PDV





**Figure II.17** : Test d'algorithme de déroulement de Flynn pour un interférogramme ondes à franges ouvertes. (a) l'interférogramme, (b) l'image de phase déroulée, (c) la mappe de qualité PDV

#### II.4.2.2 La méthode de Wei Xu (prédiction de la direction de déroulement)

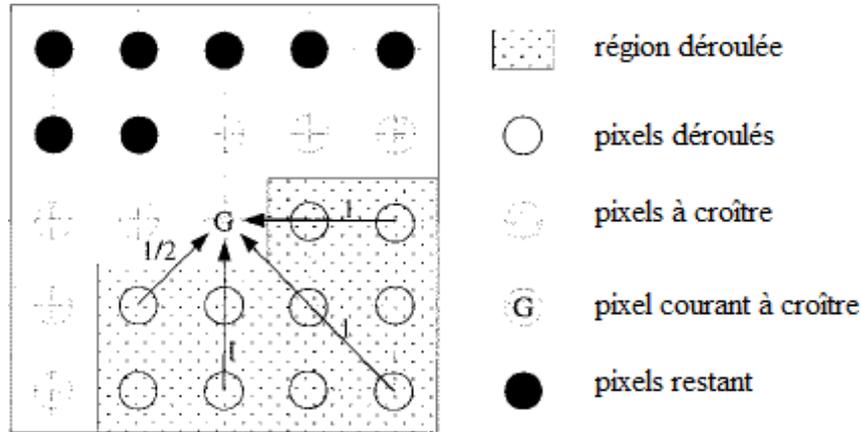
*Wei Xu* et al [35] ont proposé une autre méthode de déroulement basée également sur le processus (*RegionGrowing*), la différence entre cette méthode et celle de *Flynn* est :

- La croissance de la région se fait pour déterminer la zone à dérouler dont les pixels sont de bonne qualité (le contraire pour la méthode de *Flynn*).
- L'information de phase des pixels voisins est utilisée pour prédire la direction pertinente de déroulement de chaque pixels dans la zone saine (dans l'algorithme de *Flynn* aucune considération de l'information de phase ni dans la croissance de masque; à part la variance ou pseudo-corrélation si la cohérence n'est pas disponible, ni dans le déroulement).

L'algorithme se déroule comme suit :

- 1- La graine de départ est choisie dans l'une des régions lisses (non accidentées et non rugueuses), cela permet de faire une croissance le long des parcours contrôlée durant la procédure de déroulement.
- 2- Chaque pixel est déroulé en basant sur des prédictions déduites à partir de ses voisins déjà déroulés.
- 3- Les phases de plusieurs directions sont utilisées pour dérouler chaque pixel, cela limite l'effet de propagation d'erreur.

- 4- La vérification de fiabilité basée sur la consistance des prédictions de phase sera appliquée pour chaque tentative de déroulement afin de valider (ou de refuser) la valeur déroulée proposée.
- 5- La tolérance de fiabilité est graduellement décrétementée dans le but de permettre aux plusieurs pixels que possible d'être déroulés.



**Figure II.18** : Les prédictions des directions pour le pixel à croître et ses voisins [32]

Le déroulement de pixel à croître utilise la prédiction de phase dans une fenêtre 5 x 5 centrée sur ce pixel. Considérant les 8 voisins directs de pixel courant à croître, soit N ceux déroulés parmi ces voisins, généralement  $N > 1$  sauf pour le pixel graine. Pour avoir un déroulement plus fiable la prédiction de la phase peut être déduite à partir chacun de N voisins déroulés, ces prédictions sont calculées le long de toutes les directions possible comme il est montré dans la figure II.18, dans l'exemple de la figure ci-dessus nous avons 4 directions. La prédiction notée  $\phi_k^p$  est formée de  $k^{ième}$  voisin déroulé, elle est soit linéaire soit prédiction constante selon le nombre des pixels déjà déroulés le long de la direction :

- Si deux pixels déroulés sont disponibles le long d'une direction, une linéaire forme est utilisée  $\phi_k^p = 2\phi[k] - \phi[k']$  avec  $k'$  présente le prochain pixel déroulé dans la même direction
- Si non la prédiction est tout simplement la valeur de phase de  $k$ ,  $\phi_k^p = \phi[k]$

Puis la prédiction composite  $\phi^p$  se calcule par la moyenne pondérée de N prédictions individuelle :

$$\phi^p = \frac{(\sum_{k=1}^N w_k \phi_k^p)}{(\sum_{k=1}^N w_k)} \quad (\text{II. 18})$$

Avec  $w_k$  est un paramètre égal à 1 pour le premier cas car il est plus fiable, et égale à 0.5 pour le second tel qu'il est aussi montré dans la figure II.18. Cette prédiction de la phase est ensuite utilisée pour tentative de dérouler correctement le pixel à croître, la valeur proposée  $\phi_u = \phi_w + 2\pi m$  avec  $\phi_w$  est la phase roulée de pixel à croître et  $m$  est le nombre d'ambiguïté calculé tel que :

$$m = \text{rint} \left( \frac{\phi^p - \phi_w}{2\pi} \right) \quad (\text{II. 19})$$

Où  $\text{rint}(x)$  est l'entier le plus proche (*nearest-integer*), exemple  $\text{rint}(5.1)=5$ ,  $\text{rint}(-4.8)=-5$

A propos de la fiabilité de déroulement, plusieurs paramètres statistiques peuvent être utilisés tels que : la moyenne des déviations, la différence entre la phase déroulé du pixel et sa prédiction composite, la cohérence ou la variance. Cette mesure de fiabilité permet de valider ou de refuser la tentative de déroulement par la prédiction des différentes directions.

- Utilisant la moyenne des déviations :

$$F_p = \frac{(\sum_{k=1}^N w_k |\phi^p - \phi_k|)}{(\sum_{k=1}^N w_k)} \quad (\text{II. 20})$$

Le test de fiabilité est valide si :  $F_p < \text{un Seuil}$

Le seuil peut se décrémente avec un pas varie de  $\pi/4$  jusqu'à  $2\pi$  dans des itérations successives pour que le déroulement couvre le maximum de l'inreférogramme.

- Utilisant la différence entre la phase déroulée la prédiction composite :

$$F_p = |\phi_u - \phi^p| \quad (\text{II. 21})$$

De même la validité a lieu si  $F_p < \text{un Seuil}$ , qui se décrémente avec un pas varie de  $\pi/4$  à  $\pi$ .

- Utilisant la cohérence : mais dans ce cas la condition à vérifier est  $F_p > \text{un Seuil}$  et le pas de seuil varie de 0 à 1.
- Utilisant la variance :  $F_p > \text{un Seuil}$  et le pas de seuil varie de 0 à 2.

Cette méthode est très fiable, elle combine entre le déroulement et une sorte d'interprétation pour les zones qui auraient pu être isolées ou distordues en appliquant les méthodes précédentes (*Goldstein et Flynn*), mais très longue à exécuter.

### II.4.2.3 La méthode de quantification de la qualité

Dans la littérature [32] une méthode de déroulement hybride a été proposée, il s'agit d'une approche basée sur le même processus de croissance de la région avec des niveaux de qualité préalablement déterminés dont chaque région sera déroulée indépendamment, puis une opération de lissage entre les différentes régions sera appliquée en minimisant la moyenne carrée des phases par la méthode des éléments finies. Dans cette méthode la qualité de pixel non seulement utilisée pour le *Region-Growing* mais aussi de quantifier et diviser l'interférogramme en plusieurs régions qualifiée par un niveau de qualité.

Les étapes à suivre :

- 1- Générer la mappe de qualité. Exemple la cohérence  $\rho$
- 2- Quantification et division de l'interférogramme :

Tout d'abord nous définissons N-1 seuils dans un ordre décroissant  $S_1 > S_2 > \dots > S_{N-1}$

Nous divisons l'interférogramme en régions selon N niveaux de qualité

$$Q(m, n) = \begin{cases} 1 & |\rho| > S_1 \\ \dots \\ k & S_{k-1} \geq |\rho| > S_k \\ \dots \\ N & S_{N-1} \geq |\rho| \end{cases} \quad (\text{II. 22})$$

- 3- Dans chaque région de niveau 1 un pixel graine est sélectionné, un déroulement de phase individuelle est exécuté par le processus RG précédent (conventionnel de *Flynn* ou celui de *Wei*), puis la plus large région sera définie comme région principale M1.
- 4- Pour  $i=2$  à  $N-1$ 
  - Appliquer le RG dans la région niveau  $i$  adjacente à la principale  $M_{i-1}$ .
  - Renommer les régions de niveau  $i$  par  $i+1$ .
  - Ajuster les valeurs des pixels dans quelques régions de niveau 1 rencontrées par la région principale afin de les insérer dans cette dernière et former une région principale  $M_i$ .

Fin

- 5- Lisser les valeurs dans les régions de niveau N par la méthode des éléments finis qui consiste à minimiser :  $minL(\phi) = min\|\Delta\phi\|^2$  où  $\Delta\phi$  représente le gradient de la phase déroulée. Ajuster les valeurs des pixels dans les régions niveau 1 laissées pour les connecter à la région principale  $M_N$ .

Cette méthode est performante mais elle présente trop d'ajustements ce qui est pénible en terme du temps de traitement.

### II.4.3 Les méthodes de norme de minimisation

Dans la partie précédente nous avons présenté un état de l'art sur les méthodes de déroulement de la catégorie : suivi de parcours (*path-following*) les plus connues et utilisées comme références. Maintenant nous allons entamer la seconde catégorie : normes de minimisation (*Minimum-Norm*). Il s'agit d'autres approches qui reposent sur un concept complètement différent basées sur des minimisations des fonctions. Alors que les méthodes PF possèdent une vision locale, celle de MN possède une vision globale de déroulement en traitant le problème avec des procédures d'optimisation qui impliquent tout l'interférogramme à la fois.

#### II.4.3.1 les normes $L^p$

Les méthodes MN essaient de trouver la solution de l'image de phase absolue (déroulée)  $\psi$  pour laquelle la norme  $L^p$  de la différence (horizontale et verticale) entre la dérivée de la phase déroulée et la dérivée de la phase roulée est minimisée, car comme nous avons précédemment expliqué le gradient de phase roulée et déroulée devrait être inchangé pour le cas sans erreur.

Ce type des méthodes cherchent des images de phase déroulée dont la dérivée locale correspond à la dérivée de la phase roulée. Cette minimisation a pour le but de trouver la solution  $f$  qui peut être exprimé en générale par :

$$f = \operatorname{argmin} E(f) \quad (\text{II.23})$$

*Argmin* veut dire argument de minimisation,  $E(f)$  est la norme  $L^p$  utilisée pour la minimisation qui est :

$$E(f) = \sum_{i=1}^M \sum_{j=1}^{N-1} |\Delta^h \phi_{i,j} - \Delta^h \Psi_{i,j}|^p + \sum_{i=1}^{M-1} \sum_{j=1}^N |\Delta^v \phi_{i,j} - \Delta^v \Psi_{i,j}|^p, \quad p \geq 0 \quad (\text{II.24})$$

Avec :

$$\Delta^h \phi_{i,j} = \phi_{i,j+1} - \phi_{i,j}, \quad \Delta^v \phi_{i,j} = \phi_{i+1,j} - \phi_{i,j}$$

$$\Delta^h \Psi_{i,j} = w(\Psi_{i,j+1} - \Psi_{i,j}), \Delta^v \Psi_{i,j} = w(\Psi_{i+1,j} - \Psi_{i,j})$$

Donc la solution  $f$  conduit à une fonction  $f$  qui est la phase déroulée possible ( $\psi$ ), en plus la différence entre les termes  $\phi$  et  $\psi$  est l'entier  $k$ , alors autre manière est aussi possible en appliquant les différences sans rouler la phase  $\psi$ .

Nous pouvons aussi pondérer cette approche, si nous disposons une mappe de qualité  $Q$  en dérivant également cette mappe horizontalement et verticalement  $Q^h$  et  $Q^v$  [36].

Alors l'équation II.24 peut s'écrire :

$$E(f) = \sum_{i=1}^M \sum_{j=1}^{N-1} q_{i,j}^h |\phi_{i,j} - \Delta^h \Psi_{i,j}|^p + \sum_{i=1}^{M-1} \sum_{j=1}^N q_{i,j}^v |\phi_{i,j} - \Delta^v \Psi_{i,j}|^p, p \geq 0 \quad (\text{II. 25})$$

La mappe de qualité peut être divisée par un seuil et par conséquent elle devient binaire, donc dans les régions bruitées avec les résidus le paramètre  $q_{i,j}$  vaut 0. Et les bonnes régions se pondèrent par 1.

Ces expressions (II.24 et 25) montrent l'aspect global de cette approche le fait que toutes les phases observées sont utilisées pour calculer la solution. Les différentes valeurs de  $p$  conduisent aux différentes variantes de cette approche. Généralement les valeurs  $p \leq 2$  sont utilisées et souvent citées dans les littératures.

#### II.4.3.2 la norme $L^2$

Avec  $p=2$  nous avons le problème de moindre carré, dans ce cas nous remarquons que la minimisation, même pour le carré, est une minimisation discrète qui très demandée par les propositions récentes pourtant la majorité des algorithmes sont des solutions approximatives, l'un des inconvénients de la norme  $L^2$  est qu'elle essaie de lisser les discontinuités à moins qu'elle est employées avec pondération binaire.

*Fried et Hudgin* étaient les premiers à proposer le déroulement (avec pré-filtrage) en utilisant l'approximation moindre carrée [37,38]. Puis plusieurs algorithmes ont été publiés, parmi eux ceux qui relaxent la procédure de calcul de  $Z$  à  $R$  vue que la nature discrète du problème est plus complexe. Dans le domaine continu le problème est équivalent à résoudre équation différentielle partielle de Poisson [13]. Et cela peut être résolu en utilisant FFT ou DCT [39,40], puis revenir au domaine discret [14, 29, 30]. Une solution exacte pour la méthode de moindre carrée a été développée dans la littérature [41] en utilisant techniques de programmation réseau, et dans [42] en utilisant le max flux –min discontinuité.

### II.4.3.3 La norme $L^1$

La norme  $L^1$  est mieux que  $L^2$  en termes de préservation des franges car  $L^2$  lisse les discontinuités. Cette méthode a été proposée par Flynn [36], Costantini [43,44] en utilisant la technique de programmation réseau et [29] en utilisant max-flux dans la norme  $L^2$  et PDV comme une mappe de qualité.

### II.4.3.4 les normes de petite valeur de $p$

Avec  $0 \leq p < 1$  l'abilité de préserver les discontinuités est mieux améliorée, mais une complexité à n'est pas écarté pour ce genre de norme. En particulier  $L^0$  qui est très complexe mais en contre partie très désirable en pratique [45].

## II.5 Conclusion

Dans ce chapitre une présentation de problème de déroulement de phase, formulations pertinentes, un état de l'art sur quelques méthodes référentielles, analyse et comparaison ont été élaborés. Ce qui a permis de faire une analyse suffisante pour comprendre les mécanismes de chaque approche avec les avantages et les inconvénients.

D'après cette étude menée nous pouvons constater que les approches PF fournissent des bonnes qualités d'image de phase par rapport aux approches MN par ce que le déroulement par MN n'est qu'une estimation globale. En revanche les approches PF sont lentes et prennent plus de temps.

En comparant les méthodes BC (avec ses variantes) et QG nous pouvons constater que les méthodes BC sont plus rapide relativement aux QG mais leur inconvénient major est les régions isolées de grappe laissées sans déroulement ce qui n'est pas le cas pour les méthodes QG où toutes les régions sont déroulées en fonctions de la qualité locale. Néanmoins une hybridation entre les différentes méthodes est toujours possible afin de combiner les avantages et en sortir avec un compromis.

## **Chapitre III : Codage de canal**

### III.1 Introduction

Étant donné leur importance dans la vie quotidienne, il est évident que les communications deviennent de plus en plus cruciales. Alors, les exigences constantes en matière d'ergonomie et de qualité de service ne peuvent être satisfaites qu'en améliorant la qualité du signal. Un canal de transmission qui transmet l'information sans la modifier est nécessaire pour la communication à distance. Ce canal est sujet à un certain nombre de perturbations qui peuvent affecter la qualité du signal. Donc, nous avons tenté de couvrir l'essence du codage des canaux et la théorie des codes correcteurs d'erreurs, en particulier les codes linéaires.. Heureusement, Les codes RS de Reed Solomon, sont parmi les codes parfaits et cycliques inventé par les mathématiciens Irving S. Reed et Gustave Solomon. Il est très utilisé et permet de corriger des erreurs dues à la transition imparfaite d'un message. Pour cette raison il est très employé dans tous les domaines requérant des données fiables. [46].

### III.2 Codage de canal

Le codage de canal est une technique de traitement de signal utilisée en communication numérique pour ajouter de la redondance aux données avant leur transmission à travers un canal de communication sujet à des erreurs. Cette technique permet d'améliorer la fiabilité de la transmission en détectant et en corrigeant les erreurs qui peuvent survenir lors de la transmission. En d'autres termes, c'est le processus d'ajout de bits supplémentaires aux données avant la transmission en utilisant des algorithmes spécifiques pour l'encodage, le décodage et la correction des erreurs. Les bits redondants sont des bits supplémentaires qui sont ajoutés, et leur nombre et emplacement sont déterminés par le type de code de canal utilisé. [47] Son objectif est d'améliorer la fiabilité de la transmission des données par un canal de communication défectueux. Les erreurs peuvent être causées par divers facteurs, y compris le bruit, les interférences, l'atténuation du signal, la distorsion du signal, et ainsi de suite.

#### III.2.1 Technique de codage de canal

- **Codage de canal à redondance cyclique (CRC) :** Le CRC est une technique simple de détection d'erreurs qui consiste à ajouter des bits de contrôle à la fin du message pour permettre au destinataire de détecter les erreurs lors de la réception. Cette technique ne permet

pas de corriger les erreurs, mais elle est largement utilisée pour la détection d'erreurs dans les communications numériques.

- **Codage convolutif** : Le codage convolutif est une technique de codage de canal qui utilise des filtres pour ajouter de la redondance aux données à transmettre. Les bits de redondance sont générés à partir des bits précédents et sont ajoutés aux données à transmettre. Cette technique permet de corriger les erreurs et est utilisée dans les communications sans fil et les transmissions de données à haut débit.
- **Turbo-codage** : Le turbo-codage est une technique de codage de canal avancée qui utilise deux encodeurs convolutifs en parallèle. Cette technique permet une correction d'erreurs plus efficace que le codage convolutif standard et est utilisée dans les communications par satellite et les transmissions de données à haut débit.
- **Codage de Bose-Chaudhuri-Hocquenghem (BCH)** : Cette technique de codage de canal est utilisée pour corriger les erreurs dans les données numériques, en ajoutant de la redondance aux données à transmettre.
- **Codage de Hamming** : est un exemple de code de canal qui permet de détecter et corriger les erreurs de transmission dans les données numériques. Il s'agit d'un code de bloc, c'est-à-dire qu'il traite des blocs de données de taille fixe et ajoute de la redondance à ces blocs pour permettre la détection et la correction d'erreurs.
- **Codage de Reed-Solomon** : Cette technique de codage de canal est utilisée pour détecter et corriger les erreurs dans les données numériques, en ajoutant de la redondance aux données à transmettre.[48] qui fera l'objet de ce chapitre et qui constitue l'essentiel de ce rapport

### III.2.2 Les applications de codage de canal

De nombreuses applications de communication utilisent le codage de canal pour assurer un transfert de données fiable. Cependant, le codage des canaux comporte certaines restrictions qui peuvent limiter son efficacité dans des applications particulières. Voici quelques exemples :

- **Communications à faible débit binaire** : Le codage de canal peut nécessiter l'ajout d'une quantité importante de bits de redondance pour détecter et corriger les erreurs de transmission. Cela peut réduire le débit binaire effectif de la communication, ce qui peut être un problème dans les communications à faible débit.

- **Latence** : L'ajout de bits de redondance peut augmenter la latence de transmission des données, ce qui peut être préjudiciable dans les applications qui nécessitent une transmission en temps réel.
- **Bruit** : Les codes de canal sont efficaces pour détecter et corriger les erreurs de transmission causées par le bruit, mais ils peuvent également avoir des limitations lorsque le bruit est trop important ou provient de sources non aléatoires.
- **Bande passante** : Les codes de canal peuvent nécessiter une bande passante plus importante que la transmission de données sans correction d'erreur, ce qui peut être un problème dans les systèmes de communication à bande passante limitée. [49].

### III.3 Les codes correcteurs d'erreurs

Sont les codes utilisés pour détecter et corriger les erreurs de transmission. Ces codes sont classés en deux catégories qui ont été établies et sont actuellement utilisées sous forme de perfectionnement continu et qui sont :

- **Les codes en blocs** : (linéaires, cycliques ou non cyclique) : le codage/décodage d'un bloc dépend uniquement des informations de ce bloc.
- **Les codes en treillis** : (convolutifs, récurrents ou non) :

Le codage et le décodage d'un bloc dépendent des informations d'autres blocs, généralement des blocs qui ont été transmis auparavant. Un bref aperçu de la grande famille de codage est fourni dans la figure ci-dessous. Citons les codes les plus célèbres dans la première classe, tels que BCH, Reed Muller, Reed Solomon, Goppa, Golay et Hamming. Bien que la variété de la deuxième classe soit moins grande, elle est beaucoup plus flexible, en particulier dans le choix des paramètres et des algorithmes de décodage disponibles. Les codes convolutifs binaires systématiques récurrents sont très utilisés dans les modulations codées (TCM) et les codes concaténés parallèles (Turbo Codes) [50].

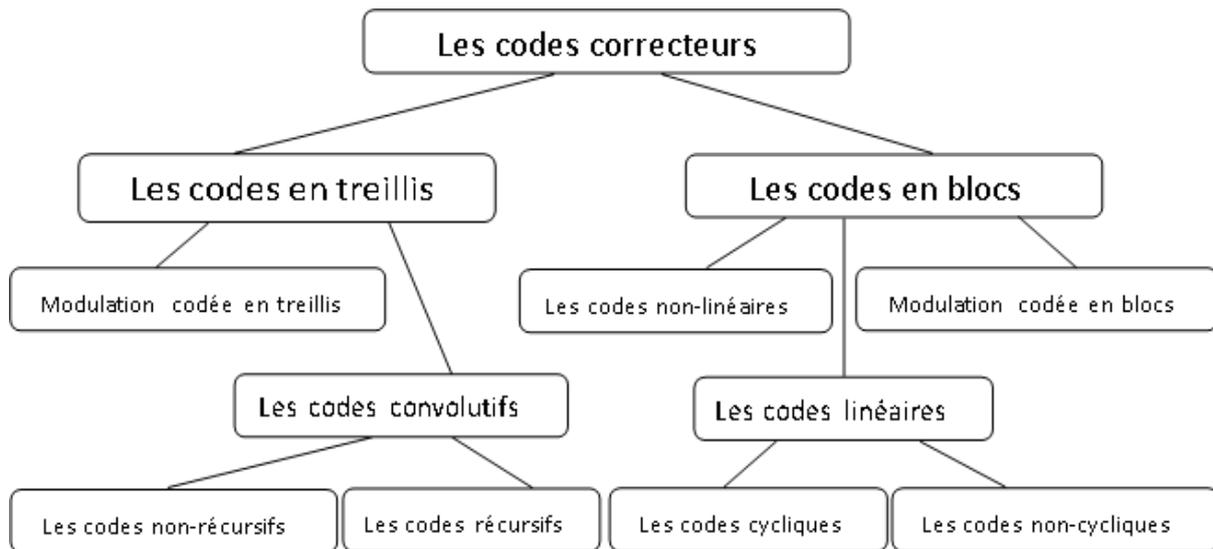


Figure. III.1 La hiérarchie des codes correcteurs [50].

### III.4 Les codes linéaires

#### III.4.1 Matrice génératrice

Un code  $\varphi$  de paramètres  $(n, k)$  est dit linéaire s'il existe une matrice  $G \in M_{n,k}(F_2)$

(C'est-à-dire avec  $n$  lignes,  $k$  colonnes, à coefficients dans  $\{0, 1\}$ ), de rang  $k$ , telle que

$$\forall m \in \{0, 1\}^k, \varphi(m) = G \times m \quad (\text{III.1})$$

La multiplication matricielle est à comprendre dans  $F_2$  (c'est à dire modulo-2 et le mot  $m$  est ici considéré comme un vecteur colonne. La condition sur le rang traduit l'injectivité de  $\varphi$ . La matrice  $G$  est appelée matrice génératrice de  $\varphi$ .

Enfin, dire que le code  $\varphi$  est systématique, c'est dire que la matrice  $G$  est de la forme  $[I_k \mid G']$  où  $I_k$  est la matrice identité d'ordre  $k$ .

#### III.4.2 Matrice de contrôle d'un code linéaire

L'intérêt principal de ne considérer que des codes linéaires est qu'ils disposent de meilleurs algorithmes de décodage. On utilise pour cela les matrices de contrôle.

- **Définition :**

Soit  $\varphi$  un code linéaire  $(n, k)$  de matrice génératrice  $G$ . On appelle matrice de contrôle de  $\varphi$  toute matrice  $H \in M_{n-k,n}$  (c'est-à-dire avec  $n$  colonnes et  $n - k$  lignes) telle que

$$H \cdot m = \vec{0} \rightarrow \Leftrightarrow m \in \mathcal{C} \quad (\text{III.2})$$

L'existence de matrices de contrôle pour n'importe quel code linéaire  $\varphi$  n'étonnera pas les spécialistes de l'algèbre linéaire :  $H$  n'est rien d'autre qu'une matrice dont le noyau est l'image de  $G$ . Il est clair que de telles matrices existent toujours.

Au passage, comme le noyau de  $H$  est  $C$ ,  $H$  contient assez d'information pour reconstituer  $\varphi$  à équivalence près.

En revanche, on remarque que  $H$  n'est pas unique en général (par exemple, en permutant deux lignes d'une matrice de contrôle, on obtient encore une matrice de contrôle).

Etant donné la matrice génératrice  $G$  d'un code  $\varphi$ , se donner une matrice de contrôle  $H$ , c'est se donner une base de l'orthogonal de  $C$  dans  $\{0, 1\}^n$ , ce qui n'est pas évident. L'opération inverse est plus facile : étant donné une matrice de contrôle  $H$ , se donner la matrice génératrice  $G$  d'un code  $\varphi$  correspondant revient à se donner une base du noyau de  $H$ .

Le théorème suivant donne un critère très simple pour déterminer une matrice de contrôle d'un code systématique [51].

- **Théorème**

Soit  $\varphi$  un code systématique de matrice génératrice  $\begin{bmatrix} I_k \\ G \end{bmatrix}$

Alors la matrice  $H = (G' I_{n-k})$  est une matrice de contrôle de  $G$ .

**Exemple**

Une matrice de contrôle du code de bit de parité (8, 9) est (1 1 1 1 1 1 1 1) [51].

### III.4.3 Distance minimale d'un code linéaire

Les codes linéaires sont également intéressants car on dispose d'informations sur leur distance minimale. Tout d'abord, par définition, dire qu'un code  $\varphi$  de paramètres  $(n, k)$  est linéaire, c'est exactement dire que  $\varphi$  est une application linéaire injective de  $\{0, 1\}^k$  dans  $\{0, 1\}^n$ .

- **Proposition (distance minimale d'un code linéaire)**

Soit  $\varphi$  un code linéaire d'image  $C$ . Alors la distance minimale de  $d\varphi$  est égale au plus petit poids non nul d'un mot de  $C$  [52].

- **Calcul de la distance minimale d'un code**

Dans le cas où  $C$  est un code linéaire binaire, on a

$$d_{min} = \min \{d(x, y) / x, y \in C, x \neq y\} \quad (III.3)$$

$$d_{min} = \min \{w(x + y) / x, y \in C, x \neq y\} \quad (III.4)$$

$$d_{min} = \min \{w(x) / x \in C, x \neq 0\} = w_{min} \quad (III.5)$$

- **Distance minimale et matrice de contrôle**

On peut encore avoir un autre critère sur la distance minimale d'un code linéaire il est donné par la matrice de contrôle du code.

- **Proposition**

Soit  $\varphi$  un code linéaire de matrice de contrôle  $H$ .

Alors  $d_\varphi$  est le nombre minimal de colonnes de  $H$  linéairement dépendantes.

- Si  $H$  a une colonne nulle, on a  $d_\varphi = 1$ .
- Sinon, et si  $H$  a deux colonnes identiques, on a  $d_\varphi = 2$ .
- Sinon, et si une colonne de  $H$  est égale à la somme de deux autres, on a  $d_\varphi = 3$ , etc.

### III.4.4 Décodage d'un code linéaire

Soit  $\varphi$  un code de paramètres  $(n, k)$ , de matrice génératrice  $G$  et de matrice de contrôle  $H$ . On se fixe un mot de source  $x$  (mot de longueur  $k$ ). Le mot de code correspondant sera  $\varphi(x) = Y$ .

Il y a éventuellement des erreurs durant la transmission et on reçoit le mot  $Z$ .

On appelle mot erreur associé à  $Z$  le mot  $E = Z - Y$ . On appelle syndrome de  $Z$  le mot  $H \times Z$ .

On note  $E$  le mot ne contenant que des 0, sauf en position  $i$ .

Quelques commentaires sur cette définition

Comme  $E = Z - Y$ , on a  $Z = Y + E$ , et donc  $E$  quantifie bien les erreurs survenues sur  $Y$  : par exemple,  $w(E)$  est le ? d'erreurs survenues. Le problème du décodage peut se reformuler : « trouver  $E$  », car alors on déduit  $Y$ .

Bien sûr, se « trouver  $E$  » est à comprendre dans le sens « trouver  $E$  le plus probable », c'est à dire « trouver  $E$  de poids minimal ».

Dire que le syndrome de  $Z$  est nul, c'est dire que  $Z$  est un mot du code, dans ce cas le mot erreur le plus probable est le mot nul [53].

## III.5 Les codes cycliques

### III.5.1 Rappels sur les polynômes

- **Définition (Polynômes à coefficients dans  $F_2$ )**

Un polynôme à coefficients dans  $F_2$  est une fonction de la forme

$$P(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n \text{ avec} \\ \forall i \in \{0, \dots, n\}, a_i \in F_2 \quad (III.6)$$

Si  $a_n \neq 0$ , l'entier  $n$  est appelé le degré du polynôme  $P$ , les entiers  $a$  sont appelés les coefficients de  $P$ , par convention le polynôme nul est considéré comme étant de degré  $-\infty$ .

- **Remarque**

Le fait de travailler dans  $F_2$  nous simplifie grandement la vie. Par exemple, on a toujours

$$(a + b)^2 = a^2 + b^2$$

- **Proposition**

Soit  $P$  un polynôme à coefficients dans  $F_2$ . Alors

$$P(x^2)P(x)^2 \quad (III.7)$$

- **Definition (racines)**

Soit  $P$  un polynôme à coefficients dans  $F_2$  et  $a \in F_2$ , On dit qu' $a$  est une racine de  $P$  lorsque  $P(a) = 0$ .

(1) Exemple :

- Le polynôme  $x^2 + x$  a pour racines 0 et 1
- Le polynôme  $x^2 + 1$  a pour racine 1.
- Le polynôme  $x^2 + x + 1$  n'a pas de racine dans  $F_2$ . Si l'on veut à tout prix qu'il ait des racines, il faudra « imaginer » de nouveaux éléments qui ne sont pas dans  $F_2$ , de la même façon qu'on construit le corps des nombres complexes en « Imaginant » un nouveau nombre  $i$  tel que  $i^2 = -1$ .

- **Définition** (factorisation, irréductibilité) Soit  $P$  un polynôme à coefficients dans  $F_2$ .

• S'il existe deux polynômes  $P_1$  et  $P_2$  tels que  $P = P_1 P_2$ , on dira que  $P_1$  et  $P_2$  divisent  $P$ , ou encore que  $P_1$  et  $P_2$  sont des diviseurs de  $P$ . Dans ce cas on a nécessairement

$$\deg(P_1) + \deg(P_2) = \deg(P) \quad (III.8)$$

S'il existe deux polynômes  $P_1$  et  $P_2$  tels que

$$\deg(P_1) \geq 1, \deg(P_2) \geq 1 \text{ et } P = P_1 P_2 \quad (III.9)$$

alors on dit que  $P_1 P_2$  est une factorisation de  $P$ .

Si  $P$  n'a pas de factorisation,  $P$  est dit irréductible

- **Proposition** (division euclidienne)

Soient  $P_1$  et  $P_2$  deux polynômes à coefficients dans  $F_2$ . Alors il existe deux polynômes à coefficients dans  $F_2$   $Q$  et  $R$ , uniques, tels que

$$P_1 = P_2 \times Q + R \text{ et } \deg(R) < \deg(P_2) \quad (III.10)$$

$Q$  est appelé le quotient de la division euclidienne de  $P_1$  par  $P_2$  et  $R$  le reste [52].

### III.5.2 Définitions d'un code cyclique

Soit  $C$  l'image d'un code de paramètres  $(n, k)$ . Le code est dit cyclique si l'ensemble des mots du code est stable par décalage circulaire. En d'autres termes, notons

$$\sigma(m_1 m_2 \dots m_{n-2} m_{n-1} m_n) = m_n m_1 m_2 \dots m_{n-2} m_{n-1} \quad (III.11)$$

Un code (d'image)  $C$  est cyclique si  $\forall m \in C, \sigma(m) \in C$ .

Exemple

- Les codes de répétition pure  $(n, k)$  sont cycliques.
- Le code par bit de parité est cyclique.

- Le code de Hamming systématique  $(7, 4)$  est cyclique.
- Plus généralement il existe des codes de Hamming  $(2^m - m - 1, 2^{m-1})$

Cycliques pour tout  $m \leq 2$ .

- Le code de Hamming étendu  $(4, 8)$  n'est pas cyclique !

### III.6 Problème principal du codage

Tout code  $C$  de paramètres  $(n, k, d)$  vérifie  $d \leq n - k + 1$ . Un code  $C(n, k, d)$  vérifiant  $d = n - k + 1$  est dit MDS (Maximum Distance Séparable). [54]

Le but du codage est de trouver le code avec le plus grand choix de mots codes de longueur déterminée avec une distance minimale donnée maximisée pour un alphabet donné  $GF(p^m)$ . Notons que dans notre utilisation des codes de Reed Solomon, nous avons des codes MDS, c'est-à-dire des codes parfaits, ou des codes qui réduisent la probabilité de confusion lors de la prise de décision ou du décodage en maximisant la distance entre les mots codes. Ces codes sont d'ailleurs appréciés pour diverses raisons. [55].

### III.7 les Corps de Galois GF (Généralités)

Il est essentiel de parler même d'une manière rudimentaire des corps de Galois abrégés en GF avant d'exposer les notions et les techniques de codage et de décodage des codes de Reed Solomon. La note qui suivra sera la suivante :

$F$  représente un corps fini (Field).

$F_p$  représente le corps fini à  $p$  éléments.

$F^*$  représente le groupe multiplicatif du corps  $F$ .

#### III.7.1 Propriétés d'un corps fini [56]

On rappelle ici, quelques propriétés utiles des corps finis. Soit  $F$  un corps fini de caractéristique  $p$  ayant  $q = n + 1$  éléments. Alors,

- $F$  est un  $F_p$ -espace vectoriel de dimension  $m$ , avec  $q = p^m$  ;
- Le groupe additif de  $F$  est isomorphe au groupe  $(\mathbb{Z}/p\mathbb{Z}, +)^m$  ;
- $F^*$  est cyclique d'ordre  $n = q - 1 = p^m - 1$  ;
- Tout élément  $x$  de  $F^*$  vérifie  $x^n = 1$ .

#### III.7.2 Construction d'un corps fini

La construction d'un corps fini est basée sur le théorème suivant

**Théorème [57]**

Soit  $\alpha$  un nombre algébrique sur  $F$ . Soit  $m$  le degré de son polynôme irréductible sur  $F$ . L'espace vectoriel engendré sur  $F$  par  $1, \alpha, \alpha^2, \dots, \alpha^{m-1}$  est alors un corps, et la dimension de cet espace vectoriel est  $m$ .

Ce théorème nous donne une méthode pour construire un corps avec une base donnée, mais ne nous donne pas de méthode pour trouver un élément primitif,  $\alpha$  n'est pas forcément un élément primitif.

**Représentation des éléments**

Il existe principalement deux méthodes pour représenter les éléments d'un corps fini qui sont

1.  $F_q$  est un  $F_p$ -espace vectoriel de dimension  $m$ , on choisit une certaine base

$\{b_1, \dots, b_m\}$  de cet espace et on repère chaque élément de  $F_q$  par ses composantes sur cette base.

2. On prend un élément  $\alpha$  de  $F^*$  qui engendre ce groupe et tout élément non nul de  $F_q$  s'écrit d'une manière, et d'une seule, sous la forme  $\alpha^l$ , avec

$$l = 0, \dots, n.$$

Pour décrire un élément du corps, on utilisera exclusivement la première représentation, car elle permet d'associer un nombre à chaque élément du corps. Mais on prendra soin de tabuler ces deux représentations, la raison en est que la représentation

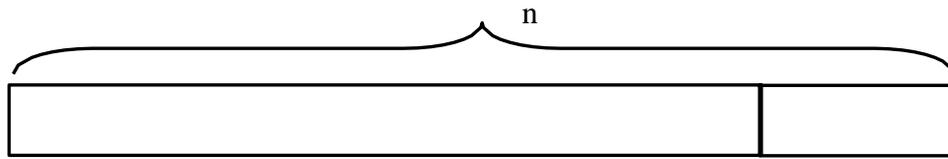
(1) se prête très bien pour l'addition et que la représentation (2) est bien adaptée pour la multiplication. Comme on a besoin de ces deux opérations, on travaillera avec ces deux représentations qui seront tabulées une fois pour toute.

Avertissement : On ne s'intéressera, par la suite, que de corps finis de caractéristique 2, qui sont les seuls utilisés pour les codes de Reed Solomon.

**III.8 Codage des codes de Reed Solomon****III.8.1 Définition :**

Reed Solomon sont des codes linéaires cycliques non binaires. Les codes RS sont non binaires lorsque les symboles n'appartiennent pas à  $GF(2)$  mais à  $(2^m)$ . Un code RS est noté  $RS(n, k)$  ou  $RS(n, k, t)$ , où  $k$  est le nombre de symboles du mot d'information et  $n$  est le nombre de symboles du mot-code et  $t$  représente le nombre maximum de symboles d'erreurs que ce code sera capable de corriger (voir la figure III.2). Pour un code  $RS(n, k)$  on définit la classe du code  $m$ , qui est le nombre de bits nécessaires pour représenter les symboles.  $n$  est relié à la classe du code par  $n = 2^m - 1$ . De plus, le nombre de symboles redondants  $n - k$  est lié au

nombre maximum d'erreurs corrigibles  $t$  par  $n - k = 2t$  [58].



**Figure. III.2** Mot-code d'un code de Reed Solomon  $RS(n,k)$ .

### III.8.2 Avantages des codes de Reed Solomon

- Ils peuvent être combiné ou ajouté à autres codes afin de réaliser des codes plus efficaces.
- L'encodage est assez facile.
- Ils sont très efficaces pour la correction d'erreurs consécutives, qu'ils soient utilisés seuls ou en conjonction avec d'autres codes. Bien entendu, cela est valide pour les cas où le nombre d'erreur demeure en dessous de la capacité de correction du code employé.
- Leurs algorithmes de décodages sont très développés. [56] [57]

### III.8.3 Principe de Codage Reed Solomon

L'équation clé définissant le codage systématique de Reed – Solomon ( $n,k$ ) est :

$$c(x) = I(x) x^{(n - k)} + [I(x) x^{(n - k)}] \text{mod}(g(x)) \quad (III.12)$$

Avec :

$C(x)$  : polynôme du mot-code, degré  $n-1$

$i(x)$  : polynôme d'information, degré  $k-1$

$[i(x)x^{n-k}] \text{mod}(g(x))$  : Polynôme de contrôle, degré  $n-k-1$

$G(x)$  : polynôme générateur, degré  $n-k$

Le codage systématique signifie que l'information est codée dans le degré élevé du mot code et que les symboles de contrôle sont introduits après les mots d'information.

#### III.8.3.1 Polynôme générateur

Les polynômes générateurs sont utilisés pour générer des symboles de contrôle. Tous les codes Reed-Solomon sont valides si et seulement si  $c(x)$  est divisible par  $g(x)$ . Un polynôme générateur de puissance  $\alpha^{2t}$  doit être utilisé pour générer un correcteur d'erreurs de symboles  $t$ . La distance minimale du polynôme est utilisée pour déterminer sa puissance maximale, qui est

$$d_{min} = 2t + 1 \quad (III.13)$$

On devrait avoir

$2t + 1$  Terme du polynôme générateur

Le polynôme générateur est sous la forme :

$$g(x) = (x - \alpha^1)(x - \alpha^2) \dots (x - \alpha^{2t}) \quad (III.14)$$

$$g(x) = (x - \alpha^1)(x - \alpha^2) \dots (x - \alpha^{2t}) \quad (III.14)$$

$$g(x) = g_{2t}x^{2t} + g_{2t-1}x^{2t-1} + \dots + g_1x + g_0 \quad (III.15)$$

### Exemple1 : calcul manuel des coefficients du polynôme générateur pour RS(15,9)

$n=15$  ;  $k=9$

Donc  $2t=n-k=15-9=6$

D'où le code Reed Solomon détectera 6 erreurs et corrigera  $t=6/2=3$  erreurs.

$$m = \ln(n+1)/\ln(2) = \ln(16)/\ln(2) = 4$$

donc on aura  $m=4$  bits par symbole

En développant la formule générale du polynôme générateur dans notre cas on obtient

$$g(x) = (x - \alpha^1)(x - \alpha^2) \dots (x - \alpha^{2t})$$

$$= (x - \alpha^1)(x - \alpha^2)(x - \alpha^3)(x - \alpha^4)(x - \alpha^5)(x - \alpha^6)$$

$$= (x^2 + \alpha^5x + \alpha^3)(x^2 + \alpha^7x + \alpha^7)(x^2 + \alpha^9x + \alpha^{11})$$

$$= (x^4 + \alpha^{13}x^3 + \alpha^6x^2 + \alpha^3x + \alpha^{10})(x^2 + \alpha^9x + \alpha^{11})$$

$$= x^6 + x^5(\alpha^{13} + \alpha^9) + x^4(\alpha^6 + \alpha^7 + \alpha^{11}) + x^3(\alpha^3 + 1 + \alpha^9) + x^2(\alpha^{10} + \alpha^{12} + \alpha^2) + x(\alpha^4 + \alpha^{14}) + \alpha^6$$

$$= x^6 + \alpha^{10}x^5 + \alpha^{14}x^4 + \alpha^4x^3 + \alpha^6x^2 + \alpha^9x + \alpha^6$$

En prenant l'équivalence en décimal, on obtient :

$$g(x) = x^6 + 7x^5 + 9x^4 + 3x^3 + 12x^2 + 10x + 12$$

### Exemple 2 : Calcul manuel des coefficients du polynôme générateur pour RS(15,11) :

$n=15$  ;  $k=11$

Donc  $2t=n-k=15-11=4$

D'où le code Reed Solomon détectera 4 erreurs et corrigera  $t=4/2=2$  erreurs.

$$m = \ln(n+1)/\ln(2) = \ln(16)/\ln(2) = 4$$

Donc on aura  $m=4$  bits par symbole.

En développant la formule générale du polynôme générateur dans notre cas on obtient :

$$g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^3)(x - \alpha^4)$$

$$= (x^2 - \alpha^2x - \alpha^1x + \alpha^3)(x - \alpha^3)(x - \alpha^4)$$

$$= (x^2 - (\alpha^2 + \alpha)x + \alpha^3)(x - \alpha^3)(x - \alpha^4)$$

$$= (x^2 - \alpha^5x + \alpha^3)(x - \alpha^3)(x - \alpha^4)$$

$$= (x^3 - \alpha^3x^2 - \alpha^5x^2 + \alpha^8x + \alpha^3x - \alpha^6)(x - \alpha^4)$$

$$\begin{aligned}
 &= (x^3 - \alpha^{11}x^2 + \alpha^{13}x - \alpha^6)(x - \alpha^4) \\
 &= x^4 - \alpha^4x^3 - \alpha^{11}x^3 + \alpha^{15}x^2 + \alpha^{13}x^2 - \alpha^{17}x - \alpha^6x + \alpha^{10} \\
 &= x^4 - (\alpha^4 + \alpha^{11})x^3 + (\alpha^{15} + \alpha^{13})x^2 - (\alpha^{17} + \alpha^6)x + \alpha^{10} \\
 &= x^4 - 13x^3 + 12x^2 - 8x + 7
 \end{aligned}$$

### III.8.3.2 Schéma général du codage Reed Solomon

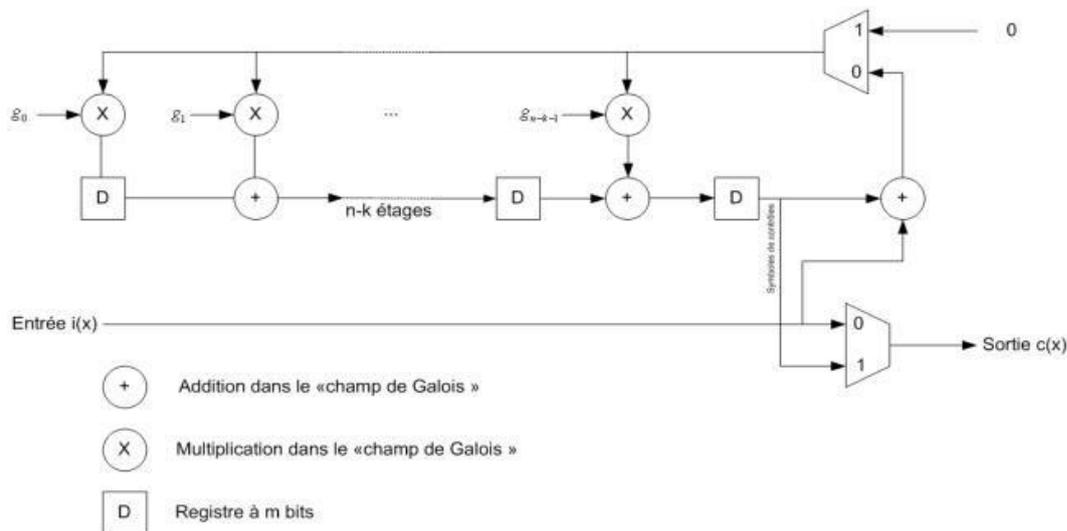


Figure III.3 : schéma de codage [59]

#### Principe du schéma :

\*initialement on remet tous les registres à 0

\*Pendant k coups d'horloge le multiplexeur 1 laisse passer l'information directement à la sortie. En même temps le circuit 1 permet de calculer les symboles de contrôle.

#### Les éléments utilisés pour le codage sont:

- Des additionneurs dans le « champ de Galois »  $GF(2^m)$
- Des multiplicateurs dans le « champ de Galois »  $GF(2^m)$
- Des multiplexeurs
- Des registers

### III.8.4 Décodage des codes Reed Solomon

L'idée du décodeur de Reed – Solomon est de détecter une séquence erronée avec peu de termes, qui sommée aux données reçues, donne lieu à un mot-code valable

les étapes nécessaires pour le décodage de ces codes :

- Calcul du syndrome
- Calcul des polynômes de localisation des erreurs et de d'amplitude
- Calcul des racines et évaluation des deux polynômes

- Somme du polynôme constitué et du polynôme reçu pour reconstituer l'information de départ sans erreur.

Considérons un code de Reed Solomon  $c(x)$  correspondant au code transmis, et soit  $d(x)$  le code que l'on reçoit. Le code d'erreur est défini par

$$e(x) = d(x) - c(x) = d(x) + c(x) \quad (III.16)$$

Car le (-) et le (+) sont équivalamment dans  $GF(2^m)$

### III.8.4.1 Calcul du syndrome :

Le calcul du syndrome peut être attribué progressivement. Attendez que tous les éléments du polynôme  $r(x)$  soient obtenus avant de pouvoir calculer le polynôme du syndrome.

Le reste indique des erreurs. On cherche une autre méthode pour calculer le syndrome car la division est toujours une opération complexe par rapport aux sommes et aux additions. [58]

Le calcul du syndrome peut aussi être effectué par un processus itératif. Avant de pouvoir calculer le polynôme du syndrome, on doit attendre que l'on ait reçu tous les éléments du polynôme  $r(x)$ .

$r(x)$  : le code que l'on reçoit. Et  $T(x)$  : le code transmis

$$S_i = r(d) \quad (III.17)$$

Donc la forme polynomiale du syndrome est la suivante :

$$S(x) = \dots S_{2t+1} x^{2t} + S_{2t} x^{2t-1} + \dots + S_2 + S_1 \quad (III.18)$$

**Remarque :** Si le code reçu n'est pas affecté par des erreurs en conséquent les coefficients du syndrome seront nuls.

➤ **Exemple :** code RS (15,9) donc  $15-9=6$  symboles de contrôle ( $2t=6$ ). Si on reçoit le polynôme suivant :

$$r(x) = \alpha x^{14} + \alpha^2 x^{12} + \alpha^{13} x^4$$

Le calcul du syndrome :

$$S_1 = r(x = \alpha) = \alpha(\alpha^{14}) + \alpha^2(\alpha^{12}) + \alpha^{13}(\alpha^4)$$

$$S_1 = r(x = \alpha) = \alpha(\alpha^{14}) + \alpha^2(\alpha^{12}) + \alpha^{13}(\alpha^4) = \alpha^{1+14} + \alpha^{2+12} + \alpha^{13+4} = 1 + \alpha^3 + 1 + \alpha^2$$

$$= \alpha^{1+14} + \alpha^{2+12} + \alpha^{13+4} = 1 + \alpha^3 + 1 + \alpha^2$$

$$= \alpha^3 + \alpha^2 = \alpha^6$$

$$S_2 = r(x = \alpha^2) = \alpha((\alpha^2)^{14}) + \alpha^2((\alpha^2)^{12}) + \alpha^7((\alpha^2)^4) = \alpha^7$$

$$S_3 = r(x = \alpha^3) = \alpha((\alpha^3)^{14}) + \alpha^2((\alpha^3)^{12}) + \alpha^7((\alpha^3)^4) = \alpha^{12}$$

$$S_4 = r(x = \alpha^4) = \alpha((\alpha^4)^{14}) + \alpha^2((\alpha^4)^{12}) + \alpha^7((\alpha^4)^4) = 0$$

$$S_5 = r(x = \alpha^5) = \alpha((\alpha^5)^{14}) + \alpha^2((\alpha^5)^{12}) + \alpha^7((\alpha^5)^4) = \alpha$$

$$S_6 = r(x = \alpha^6) = \alpha((\alpha^6)^{14}) + \alpha^2((\alpha^6)^{12}) + \alpha^7((\alpha^6)^4) = \alpha^8$$

### III.8.4.2 Évaluation du polynôme de locations d'erreurs

$M$  est l'ensemble des positions d'erreurs, on définit le polynôme de localisation par

$$l(x) = \prod_{\text{dans } M} (1 - \alpha^{p_i} \cdot x) \text{ avec } p_1, \dots, p_j, \quad (III.19)$$

Si  $b = \alpha^i$  est une racine de  $l(x)$ , donc  $l(b) = 0$ , on aura alors un des monômes qui s'annule, soit

$$(1 - \alpha^{p_i} \cdot b) = 0 \quad (III.20)$$

ce qui nous donne la relation suivante

$$b = \alpha^{-p_i} = \alpha^{q-1-p_i} \cdot \alpha^{q-1} = \alpha^i \text{ car } \alpha^{q-1} = 1 \quad (III.21)$$

La position de l'erreur sera alors donnée par  $p_i = q - 1 - i$  qui est positif et que l'on connaît, puisque l'on connaît  $b = \alpha^i$ , racine de  $l(x)$ . Il nous faudra donc chercher toutes les racines de  $l(x)$ . Il ne faut pas oublier que  $\alpha$  est un élément primitif, et que tout élément de  $\text{GF}(2^m)^*$  s'exprime comme une puissance de l'élément primitif, qui est un générateur du groupe  $\text{GF}(2^m)^*$

### III.8.4.3 Equation fondamentale

Il nous reste encore à calculer  $l(x)$ . Pour cela, on fait usage de la relation fondamentale suivante

$$l(x) \cdot S(x) = w(x) + u(x) \cdot x^{2t} \quad (III.22)$$

Comme l'on travaille dans  $(2^m)$ , on peut l'écrire sous la forme

$$l(x) \cdot S(x) + u(x) \cdot x^{2t} = w(x) \quad (III.23)$$

On peut voir cette équation comme la relation de Bézout avec des nombres, soit  $v \cdot b + u \cdot a = w$  où  $a$  et  $b$  sont des nombres donnés, et  $w$  est le plus grand diviseur commun des nombres  $a$  et  $b$ .  $u$  et  $v$  sont deux nombres qui satisfont cette relation. Notre équation ci-dessus est identique, sauf qu'elle s'applique à des polynômes.

On pose  $b = S(x)$ ,  $a = x^{2t}$

et on applique l'algorithme d'Euclide pour trouver  $(x), l(x), w(x)$ , et les racines de  $l(x)$  sont utilisées pour trouver la position de l'erreur.

Soit  $b_i$  une racine de  $l(x)$ , la valeur ou l'amplitude de l'erreur à la position  $p_i$  se calcule de la manière suivante

$$e_i = w(b_i) / l'(b_i) \quad (III.24)$$

Où  $l'(x)$  est la dérivée du polynôme  $l(x)$ .

En fait, l'algorithme d'Euclide ne donne pas  $l(x)$ ,  $v(x)$ ,  $w(x)$  mais  $Kl(x)$ ,  $Kv(x)$ ,  $Kw(x)$ , c'est-à-dire à une constante  $K$  près. Mais pour trouver les racines de  $l(x)$ , qu'on ait  $l(x)$  ou  $Kl(x)$  cela nous donne le même résultat, et comme la valeur ou l'amplitude de l'erreur est calculée par le rapport  $Kw(b_i) / Kl'(b_i)$ , on obtient le bon résultat. Il y a plusieurs méthodes de calcul des polynômes de localisation des erreurs et de d'amplitude, dans le cadre de ce mémoire on ne traitera que deux méthodes, le décodage selon l'algorithme d'Euclide et le décodage selon l'algorithme de Reed Solomon. Une fois les polynômes calculés en utilisant l'algorithme de Horner, on calculera les valeurs à soustraire pour obtenir le mot-code sans erreur [61][62].

#### III.8.4.4 Algorithme d'Euclide

On rappelle ici, le principe de l'algorithme d'Euclide appliqué à l'équation

$$v \cdot b + u \cdot a = w \text{ (III.25)}$$

##### 1) Initialization

$$q_0 = - ; \quad r_0 = a ; \quad u_0 = 1 ; \quad v_0 = 0$$

$$q_1 = - ; \quad r_1 = b ; \quad u_1 = 0 ; \quad v_1 = 1$$

##### 2) Calcul de q

On divise  $r_{i-1}$  par  $r_i$ , ce qui nous donne un quotient  $q$  et un reste  $r$ , soit  $r_{i-1} = qr_i + r$

et on pose  $q_{i+1} = q$

##### 3) Calculs

$$r_{i+1} = r_{i-1} - q_{i+1} \cdot r_i$$

$$u_{i+1} = u_{i-1} - q_{i+1} \cdot u_i$$

$$v_{i+1} = v_{i-1} - q_{i+1} \cdot v_i$$

##### 4) Test

Si  $r_{i+1}$  est différent de 0, on incrémente  $i$  et on retourne au point 2), et si  $r_{i+1} = 0$  l'algorithme est terminé. On a alors,

a)  $r_i$  est le plus grand diviseur commun entre  $a$  et  $b$ .

b)  $r_i = v_i \cdot b + u_i \cdot a$ .

En fait, la relation b) est valable pour tout  $i$  [63].

#### III.8.4.5 Algorithme de Reed Solomon

L'algorithme de décodage des codes de Reed Solomon sera le suivant

1) Calcul du polynôme syndrome  $S(x)$ .

Si  $S(x) = 0$ , il n'y a pas d'erreurs : STOP.

2) On applique l'algorithme d'Euclide avec

$$a(x) = x^{2t} \text{ et } b(x) = S(x) \quad (\text{III.26})$$

L'algorithme se termine dès que le degré de  $r_i(x)$  est inférieur à  $t$ .

Si  $r_i(x) = 0$ , il y a plus que  $t$  erreurs : STOP.

3) On pose  $L(x) = Kl(x) = v_i(x)$ .

On cherche toutes les racines de  $l(x)$ :  $b_1, \dots, b_r$ .

4) Pour chaque racine  $b_i = \alpha^j$ , la position  $p_i$  de l'erreur est

$$p_i = q - j - 1 = 2^m - j - 1 \quad (\text{III.27})$$

5) On pose  $w(x) = Kw(x) = r_i(x)$ .

La valeur de l'erreur à la position  $p_i$  est donnée par

$$e_i = W(b_i) / L'(b_i) = K \cdot w(b_i) / K \cdot l'(b_i) = w(b_i) / l'(b_i) \quad (\text{III.28})$$

La nouvelle valeur à la position  $p_i$  sera alors  $c_i = d_i + e_i$ . On évalue toujours un polynôme ainsi que sa dérivée à l'aide d'un schéma d'Horner, et ceci afin de minimiser le nombre de calculs [64].

#### III.8.4.6 Schéma d'Horner

On donne ici, le schéma d'Horner pour évaluer un polynôme et sa dérivée à une valeur donnée.

$$\text{Soit } p(x) = a_n \cdot x_n + a_{n-1} \cdot x_{n-1} + \dots + a_1 \cdot x + a_0 \quad (\text{III.29})$$

$$p'(x) = a_n \cdot x_n + a_{n-1} \cdot x_{n-1} + \dots + a_1 \cdot x + a_0 \quad (\text{III.29})$$

On désire évaluer notre polynôme et sa dérivée à  $x = \beta$ , soit  $(\beta)$  et  $p'(\beta)$ .

1) **Initialization**

$$b_n = a_n$$

$$a_n = b_n$$

$$k = n$$

2) **Itération**

$$b_k = a_k + \beta \cdot b_{k+1}$$

$$\text{Si } > 0 \text{ alors } c_k = b_k + \beta \quad c_{k+1}$$

3) **Contrôle**

Si  $k$  est différent de 0, on décrémente  $k$  et on retourne aupoint II, sinon l'algorithme est terminé.

On aura alors  $(\beta) = b_0$  et  $p'(\beta) = c_1$  [65].

### Exemple 2.4 (exemple de décodage)

Reprenons notre exemple 2.2 ci-dessus, soit notre code de Reed Solomon

$$c(x) = 9x^{14} + 8x^{13} + 7x^{12} + 6x^{11} + 5x^{10} + 4x^9 + 3x^8 + 2x^7 + 1x^6 + 6x^5 + 5x^4 + 15x^3 + 15x^2 + 11x + 14 \quad (III.30)$$

Choisissons un polynôme d'erreur, soit  $e(x) = 7x^{11} + 10x^2$ . Notre code reçu sera alors,

$$d(x) = c(x) + e(x) \quad (III.31)$$

$$d(x) = 9x^{14} + 8x^{13} + 7x^{12} + 1x^{11} + 5x^{10} + 4x^9 + 3x^8 + 2x^7 + 1x^6 + 6x^5 + 5x^4 + 15x^3 + 5x^2 + 11x + 14$$

On calcule le syndrome en  $\alpha^i$  :

$$S(\alpha^i) = d(\alpha^i) = d(2^i) \text{ avec } i = 1, \dots, 6, \text{ ce qui nous}$$

donne le syndrome sous forme polynômiale

$$S(x) = 1x^5 + 15x^4 + 7x^3 + 8x^2 + 11 \quad (III.32)$$

On pose  $(x) = x^6$  et  $(x) = (x)$ , et on applique l'algorithme d'Euclide, cela nous donne

$$L(x) = 6x^2 + 9x + 1$$

$$W(x) = 5x + 11$$

$$U(x) = 6x$$

On cherche les 2 racines pour  $l(x)$ , on trouve

$$b_1 = 9 = \alpha^4 \text{ et } b_2 = 6 = \alpha^{13}$$

d'où l'on tire

$$p_1 = 15 - 4 = 11 \text{ et } p_2 = 15 - 13 = 2.$$

Il nous reste à trouver la valeur de l'erreur à ces positions. Pour cela, on calcule

$$e_1 = W(9) / L'(9) = 13 / 9 = 7 \quad (III.33)$$

$$e_2 = (6) / L'(6) = 12 / 9 = 10 \quad (III.34)$$

Ce qui nous donne un polynôme de correction

$$v(x) = 7x^{11} + 10x^2$$

On remarquera que l'on a  $v(x) = e(x)$ , et que si l'on effectue la correction, cela nous donne  $c'(x) = d(x) + v(x) = c(x) + e(x) + v(x) = c(x)$  et que l'on retrouve notre code de départ, ce qui bien le but recherché [66].

### III.8.4.7 Détection d'erreurs dont le nombre est supérieur à $t$

Voici les principaux contrôles qu'il faut effectuer afin de détecter un nombre d'erreurs supérieur à  $t$ .

Après avoir terminé l'algorithme d'Euclide, deux cas peuvent se présenter

- $x^t$  divise  $S(x) \Rightarrow w(x) = 0$ .
- $S(x) = w(x) \Rightarrow l(x) = 1$  et donc pas de racine.

Lors de la recherche des racines de  $l(x)$ , trois cas peuvent se présenter

- 0 est racine de  $(x)$ .
- $l(x)$  a des racines multiples.
- $l(x)$  ne se décompose pas en produit de facteur linéaire. On trouve donc moins de racine qu'il ne devrait y en avoir.

Lors de la recherche de la position de l'erreur, un cas peu se présenter

- $p < 0$  ou  $p \geq n$ .

Il est clair que lorsque l'on a un nombre d'erreurs supérieur à  $t$ , tous ces cas peuvent se présenter. Il faut donc absolument faire ces contrôles afin de détecter cette situation qui peut dès lors se présenter.

## III.9 Applications des codes Reed-Solomon :

Les codes de Reed-Solomon sont aujourd'hui dans toute application multimédia et dans toute application nécessitant une correction d'erreurs. Selon l'application on aura différents types de codes utilisés.

Les principaux domaines d'utilisations des codes de Reed – Solomon sont :

- Dans la sauvegarde de données (sauvegarde magnétique, optique, etc...)
- Dans la communication mobile et les réseaux sans fils (wireless, etc...)
- Dans les communications satellitaires
- Dans la télévision numérique et la radio diffusion numérique (DVB)
- Dans les modems ADSL et VDSL [60]

### **III.10 Conclusion**

Les codes correcteurs d'erreurs comme les codes de RS avec leur technique de codage ainsi que les techniques de décodage sont bien exposés dans ce chapitre. Nous présentons également un algorithme de décodage des codes RS à savoir, l'algorithme d'Euclide. Enfin, et pour mieux comprendre l'effet d'un code RS sur un système de communication numérique on doit étudier son pouvoir de correction d'erreurs dans ce système, soit sur le champs réel ou soit à l'aide d'une simulation. La simulation de ces codes fait l'objet du chapitre suivant.

# **Chapitre IV : analyse de code de Reed Solomon**

## IV.1 Introduction

Le codage des images de télédétection est toujours indispensable car ce type d'images contient des données sensibles. Il a pour le but pour de protéger l'image (l'information en générale) contre les erreurs de transmission qui peut altérer des bits ou des trames entières. De nos jours, le développement théorique des techniques de codage améliore en pratique les systèmes de communication notamment le numérique. Malheureusement, la nature du milieu de propagation abuse cet avantage et cela peut changer le contenu informatif dont l'image fait partie. Lors de la communication sécurisée d'images de télédétection, le codage de canal est effectué avant l'envoi à une longue distance. Ici, nous allons discuter et analyser une méthode d'encodage d'images de télédétection InSAR à l'aide du code du canal de Reed-Solomon.

Les normes de codage d'image sont conçues pour confronter la sensibilité au bruit de canal, donc pour les canaux bruyants, il est nécessaire d'étudier afin de sélectionner la méthode de codage de canal appropriée comme compromis entre la qualité de l'image et la capacité à contrôler les erreurs. Codes Reed-Solomon (RS) décrits par Reed et Solomon en 1960 [67] sont de puissants codes de correction d'erreurs et sont de plus en plus fréquemment utilisés en raison de la disponibilité des composants VLSI. Les codes RS ont une puissante capacité de correction des erreurs aléatoires et en rafale.

Le code Reed-Solomon peut être adapté aux capacités de correction d'erreurs en rafale et cela sera montré dans la contribution. Notre but est d'étudier les effets du bruit sur les performances des méthodes de codage Reed-Solomon pour différentes capacités de correction d'erreurs appliquées aux interférogrammes InSAR.

Les résultats répertoriés dans notre travail indiquent les effets des codes RS sur des images particulières telles que l'interférogramme InSAR. Sur la base de cette analyse, nous pouvons généraliser l'efficacité du tel code pour les images de télédetection. Nous allons voir si les codes RS sont vraiment adaptatifs aux intéferogrammes et offrent la meilleure qualité d'image reconstruite et déroulée.

Depuis leur découverte, les codes RS sont probablement les codes de correction d'erreurs (ECC) les plus appliqués dans de nombreux systèmes de communication et d'enregistrement numériques. Outre de nombreuses applications dans le passé, des médias numériques aux

communications sans fil ainsi que dans les systèmes de mémoire et de stockage. Les codes RS sont toujours intégrés dans les nouveaux systèmes de communication d'aujourd'hui tels que WiMAX [71][72] et DVB. En raison de leur capacité à corriger les erreurs en rafale, les codes RS sont toujours préférés aux autres codes dans les environnements où des services sensibles au retard combinés à une communication robuste deviennent nécessaires. L'adaptation [68] est un problème clé pour obtenir les meilleures performances pour les communications de données dans les canaux sans fil, avec des fluctuations dynamiques des canaux qui provoquent souvent des taux d'erreur de trame élevés. Des schémas FEC adaptatifs ont été explorés ces dernières années [69].

Dans [70], les auteurs ont présenté un nouveau codage source-canal adaptatif avec rétroaction pour la transmission progressive d'images médicales. Le système s'adapte à la fois au contenu de l'image et aux spécifications du canal et semble être convenable aussi pour les interférogramme. La longueur des codes de parité peut être ajustée automatiquement en fonction de l'emplacement des sous-blocs d'image et des caractéristiques pratiques du canal de communication pour assurer une protection adéquate des données. Dans [7], les auteurs ont développé et analysé un système de décodage adaptatif d'erreurs et d'effacements Reed-Solomon basé sur FPGA qui est optimisé pour minimiser les caractéristiques de consommation d'énergie. Dans [73], les auteurs ont conçu une nouvelle architecture pour un Reed-Solomon adaptatif. Le décodeur peut s'appliquer à un dispositif informatique mobile sensible au contexte pour prendre en charge diverses spécifications de code pour des protocoles de communication hétérogènes. Le décodeur peut faire varier la longueur du mot de code, générer des polynômes et corriger des effacements. Le décodeur peut prendre en charge toute spécification comprise entre  $(1 \leq n \leq 255)$  pour le mot de code et  $(1 \leq t \leq 255)$  pour les erreurs.

Afin de fournir un schéma adaptatif basé sur les codes de correction d'erreurs RS, nous avons analysé les performances des codes RS pour différentes capacités de correction d'erreurs et différents paramètres  $t$ . Dans ce chapitre, après une brève description du code RS et de sa discussion sur les capacités de rafale, nous analysons d'abord le taux d'erreur binaire qui est défini comme le rapport entre les bits d'information transmis erronés et les bits globaux transmis et peut être utilisé pour évaluer la fonction de performance de transmission pour le codes RS, puis sur cette base nous proposons un schéma de codage/décodage adaptatif qui maximise l'efficacité de transmission en fonction des conditions du canal (SNR reçu).

Les résultats obtenus montrent qu'avec le codage adaptatif proposé, l'efficacité de la transmission peut être considérablement améliorée. Deuxièmement, nous simulons et

discutons de la qualité de transmission d'image et comparons les performances du code RS pour différentes capacités d'erreurs. Troisièmement, nous évaluons les performances des codes RS en cascade. Quatrièmement, nous suggérons une approche adaptative de Reed-Solomon basée sur la capacité de correction d'erreurs. Enfin nous concluons notre travail et nous suggérons quelques perspectives.

## IV.2 Rappel sur le code Reed- Solomon

### IV.2.1 Description

Les codes Reed Solomon (RS) sont un sous-ensemble des codes Bose Chaudhuri-Hochquenghem (BCH) [9][10] et sont des codes en blocs linéaires[11]. Ce sont de puissants codes correcteurs d'erreurs dont les symboles sont choisis dans un corps fini (Corp Galois). Leur nature non binaire les rend particulièrement adaptés pour corriger les rafales d'erreurs. Un code Reed-Solomon est spécifié comme RS (n, k) avec des symboles à m bits. Cela signifie que le codeur prend k symboles de données de m bits chacun et ajoute des symboles de parité pour créer un mot de code à n symboles. Il existe n-k symboles de parité de m bits chacun. Un décodeur Reed-Solomon peut corriger jusqu'à t symboles contenant des erreurs dans un mot de code, où  $t = (n-k)/2$ . Si l'emplacement des erreurs de symbole est marqué comme un effacement, le décodeur RS peut corriger deux fois plus d'erreurs. Aucune erreur corrigible ne se produit lorsque plus de (t) symboles reçus sont erronés dans un bloc donné. Lorsque cela se produit, l'une des deux actions se produit au niveau du décodeur. Soit le bloc de message est reconnu comme n'étant pas corrigeable et est signalé comme tel (c'est ce qu'on appelle une erreur reconnue), soit le modèle d'erreur est supposé par le décodeur comme corrigeable, et le décodeur corrige par erreur le bloc de message entier avec le mauvais message (ceci s'appelle une erreur de décodage).

#### Le codage

Soit  $(u_0, u_1, u_2, \dots, u_{k-1})$  les symboles de données de k m bits. Ces symboles sont codés dans un mot de code  $(c_0, c_1, c_2, \dots, c_{n-1})$ . Ce processus de codage est mieux décrit en termes de polynôme de données :

$$I(x) = u_0 + u_1x + u_2x^2 + \dots + u_{k-1} x^{k-1}$$

$$C(x) = c_0 + c_1x + c_2x^2 + \dots + c_{n-1} x^{n-1}$$

$C(x)$  est un polynôme multiple de  $G(x)$ , qui est le polynôme générateur du code, qui est défini comme :

$$G(x) = \prod_{i=0}^{2t-1} (x - \alpha^{m_0+i})$$

Où  $m_0$  vaut typiquement 0 ou 1. Alors que  $2t$  puissance consécutive  $\alpha^{m_0}, \alpha^{m_0+1}, \dots, \alpha^{m_0+2t-1}$  sont les racines de  $G(x)$ . Pour tous les polynômes de mots de code  $C(x)$ . En fait, un arbitraire polynôme de degré inférieur à  $n$  est un polynôme mot de code si et seulement s'il satisfait l'équation précédente. Un codage systématique produit des polynômes de mot de code qui sont composés de données suivies de symboles de contrôle de parité, et il est obtenu comme suit.

$$Pa(x) = (x^{2t} * I(x)) \text{ mod } G(x).$$

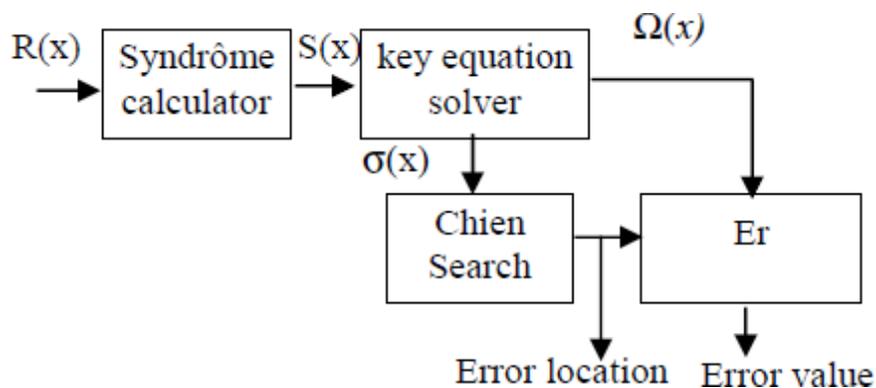
Donc, le mot de code est donné par :

$$(C_{n-1}, C_{n-2}, \dots, C_1, C_0) = (u_{k-1}, u_{k-2}, \dots, u_1, u_0, P_{n-k-1}, P_{n-k-2}, \dots, P_1, P_0)$$

Nous remarquons qu'il se compose des symboles de données suivis des symboles de vérification.

### Le décodage

Les étapes générales de décodage sont illustrées sur la figure ci-dessous.



**Figure IV.1** : Les étapes générales de décodage

Le calculateur de syndrome génère un ensemble de syndromes à partir du polynôme de mot de code reçu  $R(x)$ . À partir des syndromes, le résolveur d'équation clé produit le polynôme localisateur d'erreur  $\sigma(x)$  et le polynôme évaluateur d'erreur  $\Omega(x)$  qui peuvent être utilisés par la recherche de « Chien » et l'évaluateur de valeur d'erreur pour déterminer les emplacements exactes d'erreur et les valeurs d'erreur, respectivement.

### IV.2.2 Analyse du taux de code RS

Avant d'aller plus loin, nous présentons quelques résultats numériques et améliorons l'analyse des performances de RS. Analysons d'abord la relation entre le taux de code RS et la capacité de correction des erreurs. Le taux de code (efficacité) d'un code est donné par : taux de code =  $R = k/n$ . Les codes avec des taux de codage élevés sont généralement souhaitables car ils utilisent efficacement le canal disponible pour la transmission d'informations. Dans cette partie, nous avons classé les ensembles de codes RS ( $n < 255$ ) en fonction de leur taux de codage ; La classification des codes RS est présentée dans le tableau ci-dessous.

Pour le même taux de code, plus le mot de code est long, meilleures sont les performances, car la capacité de correction d'erreurs est augmentée. Par conséquent, si la longueur du mot de code n'est pas limitée à une certaine petite valeur, un meilleur résultat peut être obtenu. Il est utile de noter que pour une même longueur de mot de code, on ne peut pas améliorer les performances en augmentant simplement la capacité de correction d'erreurs. Lorsque la capacité de correction d'erreur dépasse une certaine valeur, le taux de code réduit n'est pas compensé par la capacité de correction d'erreur accrue du code RS.

Code specification	CODE RATE	Code specification	CODE RATE
0% < Code rate <25		50% < Code rate <75	
(7,1,3) GF(8)	14%	(7,4,1) GF(8)	57%
(15,1,7) GF(16)	6%	(15,11,1)	73%
(31,6,7) GF(32)	19%	(31,16,3)	51%
(31,1,15) GF(32)	03%	(31,21,2)	67%
(63,10,13)	15%	(63,36,5)	57%
(63,7,15)	11%	(63,39,4)	61%
(63,1,31)	02%	(63,45,3)	71%
(127,1,63)	1%	(127,64,10)	51%
(127,8,31)	06%	(127,71,9)	55%
(127,15,27)	11%	(127,78,7)	61%
(127,22,23)	17%	(127,85,6)	66%
(127,29,21)	22%	(127,92,5)	72%
25% < Code rate <=50		75% < Code rate <100	
(15,5,3)	33%	(31,26,1)	83%
(15,7,2)	46%	(63,51,2)	80%
(31,11,5)	35%	(63,57,1)	90%
(63,16,11)	25%	(127,106,3)	83%
(63,18,10)	28%	(127,113,2)	89%
(63,24,7)	38%	(127,120,1)	94%
(63,30,6)	47%		
(127,36,15)	28%		
(127,43,14)	34%		
(127,57,11)	45%		

**Tableau IV.1** : le code rate par rapport au code spécification

## Résultats

L'une des mesures de qualité d'image largement utilisée est le MSE (Erreur Quadratique moyenne). Il s'agit de la somme des écarts au carré entre les pixels correspondants de deux images (originale et reçue). Des petite valeurs MSE indiquent une forte corrélation avec la séquence d'origine et, par conséquent, l'interférogramme codé/décodé est considérée comme étant de bonne qualité.

Les résultats de simulation qui illustrent le MSE ont été obtenus pour différents capacité de correction basé sur le paramètre  $t$  et différentes longueurs du mot-code  $n$ . ceci permet de mettre en évidence comment les erreurs affectent subjectivement les images ; les petites valeurs de MSE présentent la bonne correction d'erreurs.

Pour  $n=15$

Err % \ t	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0
5	0.8	0.8	0.7	0	0	0	0	0
10	1	0.8	0.5	0.4	0	0	0	0
15	1	0.4	0.4	0.2	0.1	0	0	0
20	1	1	0.9	0.2	0.1	0.1	0	0

MSE normalisé ( $n=15$ )

**Tableau IV.2** : le MSE avec longueurs du mot-code  $n=15$

Pour  $n=63$

Err % \ t	5	10	15	20	25	30	35	40
0	0	0	0	0	0	0	0	0
5	0.5	0.5	0	0	0	0	0	0
10	0.5	0.5	0.5	0	0	0	0	0
15	1	0.5	0.4	0	0	0	0	0
20	1	1	0.7	0.4	0	0	0	0

MSE normalisé ( $n=15$ )

**Tableau IV.3** : le MSE avec longueurs du mot-code  $n=63$

### **IV.3 Conclusion**

Dans ce chapitre, nous avons analysé les performances codage RS en termes d'un paramètre largement utilisé dans le domaine de traitement d'image qui est le MSE. L'analyse est faite aussi selon différentes valeurs de  $t$  et de taux d'erreur. D'après les résultats obtenus, nous affirmons que les codes RS sont également très efficaces pour les interférogrammes InSAR comme les images ordinaires.

# *Conclusion Générale*

# Conclusion générale

En conclusion, ce mémoire a exploré l'application du codage Reed-Solomon dans le domaine des interférogrammes InSAR, visant à améliorer la fiabilité des mesures des déformations de la surface terrestre. Les résultats obtenus ont démontré l'efficacité du codage Reed-Solomon pour détecter et corriger les erreurs introduites lors de la transmission, du stockage ou du traitement des interférogrammes.

L'InSAR est une technique précieuse pour surveiller les mouvements de la surface terrestre, offrant des informations cruciales pour la compréhension des phénomènes géologiques et environnementaux. Cependant, les données InSAR sont sensibles aux perturbations et aux erreurs, ce qui peut compromettre la précision des mesures.

Le codage Reed-Solomon s'est avéré être une solution efficace pour améliorer la robustesse des données InSAR. En ajoutant des codes correcteurs d'erreurs, il est possible de détecter et de corriger les erreurs, assurant ainsi l'intégrité des données et augmentant la précision des mesures des déformations de la surface terrestre.

Ce mémoire a présenté les principes fondamentaux du codage Reed-Solomon, en soulignant son application spécifique à l'InSAR. Les expérimentations réalisées ont confirmé les avantages du codage Reed-Solomon, en termes de réduction des erreurs et d'amélioration des mesures.

Il est important de souligner que l'implémentation réussie du codage Reed-Solomon dans les interférogrammes InSAR nécessite une compréhension approfondie des paramètres et des techniques de correction d'erreurs appropriées. Une sélection judicieuse des paramètres, tels que la longueur des codes et la taille des blocs, ainsi que l'utilisation de techniques de décodage avancées, est essentielle pour maximiser les avantages du codage Reed-Solomon.

En résumé, ce mémoire a démontré que l'application du codage Reed-Solomon dans les interférogrammes InSAR constitue une approche prometteuse pour améliorer la fiabilité et la précision des mesures des déformations de la surface terrestre. Ces résultats ouvrent la voie à de nouvelles perspectives de recherche dans le domaine de l'InSAR et contribuent à une meilleure compréhension des processus géologiques et environnementaux qui façonnent notre planète.

## Bibliographies

- [1] Keith Raney, "Radar, Altimeters", Encyclopedia of Remote Sensing Springer, Vol. 88, No. 3. pp. 525-532, 2015.
- [2] P. A. Rosen, S. Hensley, I.R. Joughin, F. Li, S. N. Madsen, E. Rodriguez, and R. M. Goldstein, "Synthetic Aperture Radar Interferometry", IEEE proceeding, Vol. 88, No 3. pp. 333-382, 2000.
- [3] Bruce Chapman and Ronald G. Blom, "Chapter 10 Synthetic Aperture Radar, Technology, Past and Future Applications to Archaeology", Mapping Archaeological Landscapes from Space, Springer Briefs in Archaeology, DOI 10.1007/978-1-4614-6074-9\_10, 2013.
- [4] R. Bamler and P. Hartzl, "TopicalReview-Synthetic aperture radar interferometry". Inverse Problems, Volume 14, Issue 4, pp. R1-R54, 1998.
- [5] Z. Lu and D. Dzurisin, "Introduction to Interferometric Synthetic Aperture Radar", InSAR Imaging of Aleutian Volcanoes, Springer Praxis Books, DOI: 10.1007/978-3-642-00348-6\_1, 2014).
- [6] S. Dupont M. Berthod, " interférométrie radar et déroulement de phase", INRIA rapport de recherche N 2344, programme 4- robotique, image et vision, Projet Pastis, 1994.
- [7] G. Franceschetti and R. Lanari. "Synthetic Aperture Radar Processing". CRC Press, 1999.
- [8] M. Soumekh. "Synthetic Aperture Radar Signal Processing with MATLAB algorithms". Wiley-Intercience, 1999.
- [9] R. Chatterjee. "Antenna Theory and Practice". New Age International Pvt Ltd, New York, 2004.
- [10] T. Bayer, R. winter, G. Schreier, "Terrain influences in SAR backscatter and attempts to their correction ", IEEE transaction on geoscience and remote sensing. Vol. 29 N 3, pp 451-462, 1991.
- [11] E. Rodriguez: "Theory and design of interferometry SAR", IEE Proceedings F - Radar and Signal Processing, Vol. 139 N 2, pp. 147 – 159, 1992.
- [12] H. Zeber: "Decorrelation interferometric Radar echoes", IEEE transaction on geoscience and remotesensing, Vol. 30 N 5, pp. 950-959, 1992.

- [13] D.C. Ghiglia and M. D. Pritt, "Twodimensional phase unwrapping: Theory, algorithms and software," Wiley, New York: p. 483, 1998.
- [14] W. Dukhopelnikova, "methods of phase unwrapping for SAR interferometry", IEEE, MSMW'04 Symposium Proceedings. Kharkov, Ukraine, June 21-26, 2004.
- [15] N. Egidi and P. Maponi, "A comparative study of twofast phase unwrapping algorithms", Applied Mathematics and Computation, Elsevier Vol. 148, pp. 599–629, 2004.
- [16] K. Itoh , "Analysis of the phase unwrapping algorithm", Applied Optics, Vol. 21, No 14, pp 2470-2470, 1982.
- [17] T. Bentahar, D. Benatia and M. Boulila, "2D Phase UnwrappingAlgorithm for Phase Image usingAdjustmentMethod", J. Automation &Systems Engineering Vol 4, No 3, pp 154-164, Sep 2010.
- [18] U. Spagnolini, P. di Milano, "2-D phase unwarpping ", Society of Exploration Geophysicists, SEG Annual Meeting, Houston, Vol 2, Texas, Nov 1991.
- [19] C. Lacombe, Kornprobst, P. ; Aubert, G. ; Blanc-Feraud, "A variationalapproach to one dimensional phase unwrapping", 16th IC Pattern Recognition Proceedings. IEEE, Vol 2, 2002.
- [20] R. M. Goldstein, H. A. Zebker, C. L. Werner, "Satellite radar interferometry: two-dimensional phase nwrapping," Radio Science, vol.23, no.4, pp 713-720, July-Aug. 1988.
- [21] W. Nan, and F. Dazheng, "inSAR phase unwrapping algorithm using branch-cut and finite element," ICSP 04 proceeding, IEEE, Sep 2004.
- [22] X. Kai, D. Ning, C. Haihua, Y. Jiancheng; L. Ye, W. Yonghui, "Analysis of the quality values of the residues in branch cut phase unwrapping algorithm", Electronics, Communications and Control (ICECC), proceeding, IEEE, Sep 2011.
- [23] Cusack, R., J. M. Huntley, and H. T. Goldstein, "Improved noise- immune phase unwrapping algorithm", Applied Optics, Vol. 34, No. 5, 781-789, 1995.
- [24] Buckland, J. R., J. M. Huntley, and S. R. E. Turner, "Unwrapping noisy phase maps by use of a minimum cost matching algorithm", Applied Optics, Vol. 34, No. 23, 5100-5108, 1995.
- [25] J. Quiroga, A. Gonzalez-Cano, and E. Bernabeu, "Stable-marriages algorithm for preprocessing phase maps with discontinuity sources", Applied Optics, vol. 34, pp. 5029-5038, IO, August 1995.

- [26] X. Feng, W. Jicang, Z. Lei and L. Xiaoling, "A new method about placement of the branchcut in two-dimensional phase unwrapping", Synthetic Aperture Radar APSAR , proceeding IEEE, Nov 2007.
- [27] R.C. Gonzalez and R.E. Woods, "Digital Image Processing" (2nd Edition), Prentice Hall, 2002.
- [28] C. Li and D. Y. Zhu, "A residue-pairing algorithm for inSAR phase unwrapping", Progress in electromagnetics research, vol 95, pp 341-354, 2009.
- [29] Y. Wang, H.F. Huang, M.Q.Wu, "A New Phase Unwrapping Method for Interferograms with Discontinuities ", Radar Conference, Proceeding IEEE, May 2014.
- [30] Y. Zhang, S. Wang, G. Ji, and Z.Dong, "An Improved Quality Guided Phase Unwrapping Method and Its Applications to MRI", Progress in electromagnetics research, vol 145, pp 273-286 , 2014.
- [31] S. W. Xiao, F. Pan, F. Wang, L. Cong Liu, "Complex-amplitude-based phase unwrapping method for digital holographic microscopy," Optics and Lasers in Engineering, Vol. 50, No. 3, pp 322-327, Mar 2012.
- [32] F. Da-zheng, W. Nan, L. Baoquan, "A Region-growing Phase Unwrapping Approach Based on Local Frequency Estimation for Interferometric SAR", 8th signal processing conference, Proceeding IEEE, Vol. 4, 2006.
- [33] T. J. Flynn, "Consistent 2-D phase unwrapping guided by a quality map,". IGARSS'96,Proceeding, Lincoln, NE, pp 2057–2059, Mar 1996.
- [34] M. Roth, "Phase Unwrapping for Interferometric SAR by the Least-Error Path", Johns Hopkins University Applied Physics Lab Technical Report, Laurel, Md, vol. 30, 1995.
- [35] W. Xu, and I. Cumming, "A Region-Growing Algorithm for InSAR Phase Unwrapping", IEEE Transactions on Geoscience And Remote Sensing, VOL. 37, No1, pp 124-134, Jan 1999.
- [36] T. Flynn. "Two-dimensional phase unwrapping with minimum weighted discontinuity", Journal of the Optical Society of America A, Vol 14, No 10, pp 2692–2701, 1997.
- [37] D. Fried. "Least-squares fitting a wave-front distortion estimate to an array of phase-difference measurements". Journal of the Optical Society of America, Vol 67, No 3, pp 370–375, 1977.
- [38] R. Hudgin. "Wave-front reconstruction for compensated imaging". Journal of the Optical Society of America, Vol 67, No 3, pp 375–378, 1977.

- [39] W. Shi, Y. Zhu, Y. Yao, "Discussion about the DCT/FFT phase-unwrapping algorithm for interferometric applications", *Optik - International Journal for Light and Electron Optics*, Vol 121, No 16, pp 1443–1449, Sep 2010.
- [40] D. Ghiglia and L. Romero. "Robust two-dimensional weighted and unweighted phase unwrapping that uses fast transforms and iterative methods". *Journal of the Optical Society of America A*, Vol 11, pp 107–117, 1994.
- [41] J. Dias and J. Leitaó. "The  $Z\pi M$  algorithm for interferometric image reconstruction in SAR/SAS". *IEEE Transactions on Image Processing*, Vol 11, No 4, pp 408–422, April 2002.
- [42] J. Dias and G. Valadao. "Phase unwrapping via graph cut". *IEEE Transactions on Image Processing*, Vol 16, No 3, pp 698–709, Mar 2007.
- [43] M. Costantini. "A novel phase unwrapping method based on network programming". *IEEE Transactions on Geoscience and Remote Sensing*, Vol 36, No 3, pp 813–821, May 1998.
- [44] M. Costantini and A. Farina. "A fast phase unwrapping algorithm for SAR interferometry". *IEEE Transactions on Geoscience and Remote Sensing*, Vol 37, No 1, pp 452–460, Jan 1999.
- [45] C. Chen and H. Zebker. "Network approaches to two-dimensional phase unwrapping: intractability and two new algorithms." *Journal of the Optical Society of America*, Vol 17, No 3, pp 401–414, Mar 2000.
- [46] Stephen B. Wicker, Vijay K. Bhargava, *Reed-Solomon Codes and Their Applications*, IEEE Press, New York 1999.
- [47] *Techniques de l'Ingénieur*. (S. d.). Documentation scientifique et technique, conseil et formation | Techniques de l'Ingénieur. <https://www.techniques-ingenieur.fr/>
- [48] Shu Lin, Daniel J. Costello, Jr., "Error Control Coding: Fundamentals and Applications", Prentice Hall, 2004.
- [49] Stephen B. Wicker, "Error Control Systems for Digital Communication and Storage", Prentice Hall, 1995.
- [50] M<sup>lle</sup>. Khadija SERIR, Mémoire de Master, thème « Application des codes correcteurs d'erreurs Reed Muller », Université Abou Bakr Belkaid–Tlemcen 2011.
- [51] A. Spătaru, *Fondements de la théorie de la transmission de l'information*, PPUR, 1987 .
- [52] Pierre Abbrugiati, *Introduction aux codes correcteurs d'erreurs*, 23 janvier 2006.

- [53] Matthieu Finiasz, Nouvelles constructions utilisant des codes correcteurs d'erreurs en cryptographie à clef Publique, 2004.
- [54] Les codes algébriques principaux et leur décodage, Daniel Augot, INRIA Saclay-Île de France et Ecole polytechnique, Journées nationales du calcul formel Luminy, Mai 2010.
- [55] Hankerson, DR et al. Théorie du codage et cryptographie: Les Essentiels. Marcel Dekker, New York, 1991.
- [56] M. Reversat et Benoît Zhang, Cours de théorie des corps, université Paul Sabatier de Toulouse, 2003.
- [57] Romain Basson, Préparation à l'agrégation (Corps finis), 2014-2015.
- [58] Nicolas BARBOT, Codage de canal pour les communications optiques sans fil, Université DELIMOGES, le 22 novembre 2013.
- [59] Self-correcting codes conquer noise, Reed – Solomon codecs, S. S. Shab, S. Yaqub, F. Suleman, designfeature
- [60] Wicker .SB, Bhargava. VK, An Introduction to Reed-Solomon Codes.
- [61] Jen-Louis, Une nouvelle méthode de décodage des Reed-Solomon, (A new méthode for décodage Reed-Solomon codes), 1982.
- [62] Samuele Dietler, Implémentation de codes de Reed-Solomon sur FPGA pour communications Spatial (Code correcteur d'erreurs), 2005.
- [63] Pierre lesanne, L'algorithme d'Euclide, 18 Janvier 2006.
- [64] Daniel Augot, Les codes algébriques principaux et leur décodage, INRIA Saclay-Île de France et Ecole polytechnique, Mai 2010.
- [65] Hall J.I , Generalized Reed-Solomon, Michigan State University.
- [66] Moon T.K, ERROR CORRECTION CODING, Mathematical Methods and Algorithms, New Jersey : John Wiley & Sons, Inc., 2005.
- [67] I. S. Reed and G. Solomon, " Polynomial codes over certain finite fields", Journal of the Society for Industrial and Applied Mathematics, 8:300-304, 1960.
- [68] Richard J. Lavery, "Throughput optimization for wireless data ", Thesis (Telecommunications Networks), Polytechnic university, June 2001.
- [69] J. C. Bolot, S. Fosse-Parisis, and D. Towsley, "Adaptive FEC-based error control for Internet telephony," Proc. of IEEE INFOCOM '99, March 1999.
- [70] Jen-Lung Lo,1 Saeid Sanei,1 and Kianoush Nazarpour, "An Adaptive Source-Channel Coding with Feedback for Progressive Transmission of Medical Images", Research Article, Hindawi Publishing Corporation, International Journal of Telemedicine and Applications, Volume 2009, Article ID 519417, 12 pages.

- [71] 802.16™ IEEE Standard , "Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems", USA, IEEE Computer Society and the IEEE Microwave Theory and Techniques Society Sponsored by the LAN/MAN Standards Committee, October 2004, Print: SH95246, PDF: SS95246
- [72] Ashraful Islam, Riaz Uddin Mondal, Md. Zahid Hasan, Performance Evaluation of Wimax Physical Layer under Adaptive Modulation Techniques and Communication Channels, (IJCSIS) International Journal of Computer Science and Information Security, Vol. 5, No.1,
- [73] à ajouter plus tard

# Glossaire

Les termes suivants, classés dans l'ordre alphabétique, sont utilisés dans le texte.

<b>AD</b>	: Appariement par <b>D</b> istance
<b>BC</b>	: <b>B</b> ranched- <b>C</b> ut
<b>BCH</b>	: <b>B</b> ose- <b>C</b> haudhuri- <b>H</b> ocquenghem
<b>CRC</b>	: <b>C</b> yclic- <b>R</b> edundancy <b>C</b> heck
<b>DCT</b>	: <b>D</b> iscrete <b>C</b> osine <b>T</b> ransform
<b>DEM</b>	: <b>D</b> EM <b>D</b> igital <b>E</b> arth <b>M</b> odel
<b>DVB</b>	: <b>D</b> igital <b>V</b> ideo <b>B</b> roadcasting
<b>ECC</b>	: <b>C</b> odes de <b>C</b> orrection d' <b>E</b> rreurs
<b>ERS-1</b>	: <b>E</b> uropean <b>R</b> emote-sensing <b>S</b> atellite 1
<b>ERS-2</b>	: <b>E</b> uropean <b>R</b> emote-sensing <b>S</b> atellite 2
<b>FFT</b>	: <b>F</b> ast <b>F</b> ourier <b>T</b> ransform
<b>FPGA</b>	: <b>F</b> ield <b>P</b> rogrammable <b>G</b> ate <b>A</b> rrays
<b>GF</b>	: <b>G</b> alois <b>F</b> ield
<b>InSAR</b>	: <b>I</b> nterferometric <b>S</b> ynthetic <b>A</b> preture <b>R</b> adar
<b>JERS-1</b>	: <b>J</b> apanese <b>E</b> arth <b>R</b> esources <b>S</b> atellite 1
<b>MCD</b>	: <b>M</b> odel <b>C</b> onceptuel des <b>D</b> onnées
<b>MDS</b>	: <b>M</b> aximum <b>D</b> istance <b>S</b> eparable
<b>MN</b>	: <b>M</b> inimum <b>N</b> orm.
<b>MNT</b>	: <b>M</b> odèle <b>N</b> umérique de <b>T</b> errain
<b>MSE</b>	: <b>M</b> ean <b>S</b> quared <b>E</b> rror
<b>PDV</b>	: <b>V</b> ariance <b>D</b> érivative de la <b>P</b> hase
<b>PF</b>	: <b>P</b> ath <b>F</b> ollowing
<b>PVD</b>	: <b>P</b> hysical <b>V</b> apor <b>D</b> isposition
<b>RS</b>	: <b>R</b> eed- <b>S</b> olomon
<b>SAR</b>	: <b>S</b> ynthetic <b>A</b> preture <b>R</b> adar
<b>SLAR</b>	: <b>S</b> ide <b>L</b> ooking <b>A</b> irborne <b>R</b> adar

**SNR** : Signal -to- Noise **R**atio

**WiMAX** : **W**orld **W**ide inter operability for **M**icrowave **A**ccess

**Résumé :**

*L'InSAR (Radar à ouverture synthétique interférométrique) est une technique de télédétection qui utilise des données radar satellites pour mesurer et surveiller les déformations du sol avec une grande précision. Elle implique l'analyse des différences de phase entre deux ou plusieurs images radar acquises à partir de positions légèrement différentes sur la même zone.*

*Le déroulement de phase est une étape cruciale dans le traitement des données InSAR. Il vise à éliminer les ambiguïtés de  $2\pi$  dans les mesures de phase causées par la nature périodique du signal de phase. Il permet d'estimer avec précision les valeurs absolues de phase, qui sont directement liées aux déformations observées de la surface.*

*Les résultats répertoriés dans notre travail indiquent les effets des codes RS sur des images particulières telles que l'interférogramme InSAR. Sur la base de cette analyse, nous pouvons généraliser l'efficacité du tel code pour les images de télédétection. Nous allons voir si les codes RS sont vraiment adaptatifs aux interférogrammes et offrent la meilleure qualité d'image reconstruite et déroulée.*

**Mots clés :** *inSAR , MNT, déroulement de phase, codage Reed Solomon ,code correcteur ,corps de Galois ,MSE.*

---

**Abstract:**

*InSAR (Interferometric Synthetic Aperture Radar) is a remote sensing technique that utilizes satellite radar data for highly precise measurement and monitoring of ground deformations. It involves analyzing the phase differences between two or more radar images acquired from slightly different positions over the same area.*

*Phase unwrapping is a critical step in the processing of InSAR data. It aims to remove the  $2\pi$  ambiguities in phase measurements caused by the periodic nature of the phase signal. It accurately estimates the absolute phase values, which are directly related to the observed surface deformations.*

*The results presented in our study indicate the impact of Reed Solomon (RS) codes on specific images, such as InSAR interferograms. Based on this analysis, we can generalize the effectiveness of these codes for remote sensing images. We will investigate whether RS codes are truly adaptable to interferograms and offer the best quality for reconstructed and unwrapped images.*

**Index term:** *InSAR, Digital Elevation Model (DEM), phase unwrapping, Reed Solomon coding, error correction code, Galois field, MSE (Mean Squared Error).*

## ملخص

(رادار الفتحة الصطناعية التداخلية) هو تقنية استشعار عن بعد تستخدم بيانات الرادار الساتلية لقياس ورصد التشوهات الأرضية بدقة عالية. يتضمن تحليل اختلافات الطور بين صورتين أو أكثر من صور الرادار المكتسبة من مواقع مختلفة

قليلاً في نفس المنطقة

وتمثل سير العمل في المرحلة خطوة حاسمة في معالجة بيانات البحث والإنقاذ. ويهدف إلى القضاء على أوجه الغموض في بي بي قياسات الطور الناجمة عن الطبيعة الدورية لإشارة الطور. إنه يجعل من الممكن تقدير قيم الطور المطلق بدقة، والتي ترتبط مباشرة بالتشوهات المرصودة للسطح.

تشير النتائج المدرجة في عملنا إلى تأثيرات رموز آر إس على صور معينة مثل مخطط التداخل انسار. واستناداً إلى هذا التحليل، يمكننا تعميم فعالية هذه الشفرة لصور الاستشعار عن بعد. سنرى ما إذا كانت رموز آر إس قابلة للتكيف حَقًا مع الانتار فيبروغرام وتقدم أفضل جودة صورة أعيد بناؤها وعدم تدريجها.

**الكلمات المفتاحية:** رادار الفتحة الصطناعية التداخلية، تغيير المرحلة، نموذج التضاريس الأرضية، ترميز ريد سولومون، كود التصحيح، حقل جالوى، معدل الخطأ التريبيعي المعطيات.