

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ MOULOU D MAMMERI DE TIZI-OUZOU

FACULTÉ DU GÉNIE DE LA CONSTRUCTION

- DEPARTEMENT GENIE MECANIQUE -



MÉMOIRE DE FIN D'ETUDES

FILIÈRE : ELECTROMECHANIQUE

SPÉCIALITÉ MAINTENANCE INDUSTRIELLE

Identification d'endommagements par la Méthode de Rapport des Changements des Fréquences Locales – Algorithmes d'optimisation.

Présenté par : **LOUNAS Amel**

Supervisé par Mr TIACHACHT Samir

2021/2022

Remerciements

Arrivant à l'aboutissement de la tâche, je me trouve dans l'obligation respectueuse de devoir présenter mes chaleureux remerciements et témoignages de mes grâtes à tous ceux qui ont contribué aimablement et avec patience à l'élaboration de ce mémoire.

Tout d'abord, je tiens à remercier profondément les membres du jury qui nous ont fait l'honneur de juger mon travail.

Je souhaite aussi remercier très sincèrement Mr TIACHACHT Samir mon encadreur, j'ai eu l'honneur d'être parmi ses étudiants et de bénéficier de son riche enseignement tout au long de mon parcours d'études. Veuillez bien monsieur recevoir mes remerciements pour votre générosité en matière d'encadrement, pour vos judicieux conseils, votre soutien moral et votre aide durant l'élaboration de ce projet.

Dédicaces

À mes chers parents

Dont le mérite, les sacrifices et les qualités humaines

M'ont permis de réussir et d'arriver jusqu'au la

Et de vivre ce jour

Je leurs souhaite de pouvoir vivre toutes nos réussites moi et mes

sœurs et une très longue vie

À mes deux sœurs pour qui je souhaite un avenir meilleur et une très

grande réussite

À mes amis et mes coaches et mon équipe sportive(ACY) et (JSK).

Table des Matières

Dédicaces.....	iv
Table des Matières	v
Table des Figures	ix
Liste des Tableaux.....	xii
Liste des Abréviations et D'acronymes	xiii
Introduction Générale	1
Chapitre I. Méthode des éléments finis et dynamique des structures.....	3
I.1 Introduction :.....	3
I.2 Méthodes des éléments finis :.....	3
I.2.1 Application de la méthode des éléments finis sur élément barre :.....	5
I.2.2 L'assemblage des matrices :	15
I.2.3 Les conditions aux limites :	15
I.2.4 Transformation des coordonnées :	15
I.3 L'analyse dynamique des structures	18
I.3.1 Equation de mouvement :	18
I.3.2 Système Masse –Ressort (non amorti):.....	20

I.3.3	Matrice de Masse :	21
I.3.4	Matrice de Masse pour une barre 1D :	22
I.3.5	Matrice de masse pour une barre 2D :	24
I.3.6	Matrice de masse pour une barre 3D :	28
I.4	Programmation sur Matlab :	31
I.4.1	Barre 1 D et 2D :	31
I.4.2	Barre 3D :	34
Chapitre II.	Méthodes de détection d'endommagement.....	36
I.1	Introduction :	36
I.2	Méthodes de détection des endommagements basées sur les vibrations :	37
I.2.1	Méthodes basées sur les changements de fréquence :	39
I.2.2	Méthodes basées sur les changements de forme de mode :	40
I.2.3	Méthode de la courbure de la forme du mode	41
I.2.4	Méthodes basées sur les changements de la matrice de flexibilité:	41
I.2.5	Méthodes basées sur l'amortissement :	42
I.2.6	Méthode basée sur le vecteur résiduel modal	43
I.2.7	Méthode de l'énergie de déformation modale :	44
I.2.8	Méthodes de recalage de la matrice :	45
I.2.9	Méthodes de réseau neuronal :	45
I.2.10	Algorithme génétique :	47
Chapitre III.	Méthode de Rapport des changements des fréquences locales.	49
I.1	Introduction.....	49
I.2	. Recherche Bibliographique sur la méthode :	49

I.3	Présentation de la méthode :	51
I.4	Programmation de la méthode :	54
I.4.1	Structure 20 barres :	54
I.4.2	Structure 28 barres :	57
I.5	Applications numériques (LFCR).....	59
I.5.1	Treillis 2 D :	59
I.5.2	Treillis 3D :	62
I.6	Conclusion sur les résultats :	64
Chapitre IV.	Méthodes d'optimisation et quantification	65
I.1	Introduction :	65
I.2	Généralités sur les algorithmes d'optimisation :	66
I.2.1	Variable du problème :	67
I.2.2	Contraintes :	68
a)	Les contraintes d'inégalité :	68
b)	Les contraintes d'égalité :	69
I.2.3	Fonctions objectives :	70
I.2.4	Limites des variables :	72
I.2.5	Les types des algorithmes d'optimisation :	73
I.2.6	Espace de recherche :	74
I.2.7	Codage binaire :	74
I.3	Les paramètres d'un algorithme d'optimisation :	76
I.4	Algorithme d'essaim de tuniciers (TSA) :	78
I.4.1	Présentation de l'algorithme TSA :	81

I.4.2	Les étapes et l'organigramme de la TSA :.....	84
I.5	Optimiseur de pissenlit (DO) :.....	88
I.5.1	Présentation de l'algorithme DO :	90
I.5.2	Les Etapes de l'algorithme DO :.....	93
I.6	Tests et résultats :.....	97
I.6.1	Barre 2D 20 éléments :.....	97
I.6.2	Barre 3 D 28 éléments :.....	113
I.6.3	Conclusion sur les tests :.....	128
Conclusion Générale.....		130
References Bibliographies		132

Table des Figures

Figure 1. Elément Barre 2 dimensions	9
Figure 2. Elément Barre a 3 dimensions	13
Figure 3. Schéma des 4 niveaux de détection des endommagements	38
Figure 4. Treillis 2D 20 éléments	60
Figure 5:Histogramme des valeurs LFCR d'endommagement de l'élément 4	60
Figure 6:Histogramme des valeurs LFCR pour l'endommagement des éléments 2,9. 61	
Figure 7:Histogramme des valeurs LFCR pour l'endommagements des éléments 8,11,12	61
Figure 8:Treillis 3 D 28 éléments	62
Figure 9:Histogramme des valeurs LFCR pour l'endommagement de l'élément 6	63
Figure 10:Histogramme des valeurs LFCR pour l'endommagement des éléments 3 et 14.....	63
Figure 11:Histogramme des valeurs LFCR pour l'endommagement des éléments 25 et 18 et 17	64
Figure 12. Processus de l'algorithme de l'optimisation.....	67
Figure 13:Les cinq niveaux d'organisation de l'algorithme d'optimisation	75
Figure 14. L'image réel des Tuniciers et leur schéma	79
Figure 15:L'image réel du Pissenlit et son schéma.....	88

Figure 16. Localisation et quantification des éléments endommagés – cas 1 – barre 2D – population 30.	99
Figure 17 : Convergence des éléments endommagés – cas 1 – barre 2D – population 30.	100
Figure 18 : Localisation et quantification des éléments endommagés -cas1-barre2D-Itération 50.....	101
Figure 19 : convergence des éléments –cas1-barre2D-Itération 50.....	102
Figure 20 : Localisation et quantification des éléments endommagés –cas2-barre 2D-population 30	104
Figure 21: convergence des éléments endommagés –cas2-barre2D-population 30..	105
Figure 22:Localisation et quantification des éléments endommagés –Cas 2-Barre 2D-itération 50	106
Figure 23:convergence des éléments endommagés –cas 2-barre 2D-iteration 50.....	107
Figure 24 :Localisation et quantification des éléments endommagés –cas3-barre2D-population 30	109
Figure 25 : Convergence des éléments endommagés-cas3- –barre 2 D –population 30	110
Figure 26 :Localisation et quantification des éléments endommagés –cas3-barre 2D-iteration 50	111
Figure 27 : Convergence des éléments endommages-cas3 –barre 2d –Itération 50..	112
Figure 28:Localisation et quantification des éléments endommagés –cas1-barre3D-population30	114
Figure 29:Convergence des éléments endommagés –cas1-barre3D-population 30..	115
Figure 30 : Localisation et quantification des éléments endommagés –cas1-barre3D-iteration 50	116
Figure 31 : Figure 30 :convergence des éléments endommagés –Cas 1 –barre 3D-	

itération 50	117
Figure 32 : Localisation et quantification des éléments endommagés –Cas2-Barre3D- population 30	119
Figure 33 :Convergence des éléments endommagés –Cas2-barre3D-population 30	120
Figure 34:Localisation et quantification des éléments endommagés –cas2-barre3d- itération 50	121
Figure 35:Convergence des éléments endommagés –Cas2-Barre3D-itération50	122
Figure 36:Localisation et quantification des éléments endommagés –Cas3-barre3D- population 30	124
Figure 37:Convergence des éléments endommagés –Cas3-Barre3D-population30 .	125
Figure 38:Localisation et quantification des éléments endommagés –Cas3-barre3D- itération 50	126
Figure 39:Convergence des éléments endommagés –Cas3-barre3D-itération50.....	127

Liste des Tableaux

Tableau 1. Pourcentage des éléments endommagés barre2D	55
Tableau 2. Comparaison des fréquences saines et endommagées barre2D	55
Tableau 3. Pourcentage des éléments endommages barre3D	57
Tableau 4 : Comparaison des fréquences saines et endommagées barre 3D	57
Tableau 5. Les temps d'exécution des algorithmes TSA et DO sur la barre 2D	97
Tableau 12. Temps d'exécution des deux algorithmes TSA et DO -Barre3D-	113
Tableau 19. Pourcentage des endommagements localisés -Barre 2D-des deux algorithmes TSA et DO.....	129
Tableau 20. Pourcentage des endommagements localisés -Barre 3D-des deux algorithmes TSA et DO.....	129

Liste des Abréviations et D'acronymes

σ	Contrainte de la structure	N	Force axiale
ρ	Densité de masse	U	Force nodale verticale
ε	Déformation de la structure	V	Force nodale horizontale
u_i	Déplacement au nœud i	ϕ	Forme modale
u_j	Déplacement au nœud j	LFu_{ij}	Fréquence saines locales aux nœuds i,j
$[\delta]^e$	Déplacement élémentaire	LFd_{ij}	Fréquence endommagées locales aux nœuds i,j
dx	Déplacement selon l'axe x	W	Fréquence naturelle
dy	Déplacement selon l'axe y	L	Longueur de l'élément barre
dz	Déplacement selon l'axe z	$\bar{m}$	Masse par unité de longueur
$[F_i]$	Force nodale au nœud i	E	Module de Young
$[F_j]$	Force nodale au nœud j	X_i^L	Optimum minimum
$[K]^e$	Matrice de rigidité élémentaire	X_i^U	Optimum maximum
$[K]$	Matrice de rigidité globale	A	Surface de la structure
$[M]$	Matrice de masse	λ	Valeur propre
$[M]^e$	Matrice de masse élémentaire		
$[C]$	Matrice d'amortissement		

ITSA	Algorithme d'essaim de tuniciers amélioré
DO	Algorithme de pissenlit
TSA	Algorithme de tunisiens
AG	Algorithme génétique
MA	Algorithme Méta heuristiques
DAPM	Algorithme de pissenlit avec mutation basée sur la probabilité
DDL	Degré de liberté
DOF	Degrees of freedom
DA	Dondelion algorithme
MSEBI	Indicateur basé sur l'énergie de déformation modale
MEF	Méthode des éléments finis
MAC	Modal Assurance Criterion
CNN	Réseaux de neurones convolution els simples
LFCR	Rapport des changements de fréquences locales
SHM	Structural Health Monitoring
LEO	Opérateur d'échappement locale
OE	Optimisation évolutionnaire
PSO	Optimisation de l'essaim de particules
NFV	Vecteur de fréquence naturelle

Introduction Générale

La tenue des structures face aux sollicitations environnementales (séismes, vents, vagues, chocs ...) reste une problématique majeure à la fois pour la sécurité des personnes mais aussi pour la pérennité des ouvrages. Toutes les structures, telles que les ponts et les bâtiments, se détériorent avec le temps, les raisons de cette détérioration sont ces effets de l'environnement. L'inspection visuelle est la principale méthode pour maintenir la sécurité de ces structures et cette méthode consiste à une inspection planifiée sur le terrain, généralement une fois tous les plusieurs années. Cependant, cette inspection est limitée de précision et d'efficacité, de graves endommagements peuvent survenir sur les structures entre les périodes d'inspection, et il n'est pas facile d'évaluer les conditions structurelles après des événements majeurs. Par conséquent, L'autre voie majeure est l'analyse numérique. En effet, la puissance de calcul actuelle permet de modéliser et de simuler des ouvrages aussi grands et complexes. Les méthodes d'analyse numérique des structures sont aussi nombreuses que variées : éléments finis, volumes finis, éléments discrets. La formation d'un ingénieur est le développement des compétences nécessaires à l'analyse et à la prévision du comportement des systèmes d'ingénierie dans différentes conditions de charge. Seule une petite partie des problèmes réels d'ingénierie peuvent être résolus de manière analytique, d'où la nécessité d'utiliser des méthodes numériques capables de simuler avec précision des phénomènes. La méthode des éléments finis est l'une de ces méthodes numériques largement utilisées.

La surveillance de la santé des structures (SHM) fournit des outils importants elle pourrait inclure un contenu plus quantitatif que celui que l'inspection visuelle peut fournir. L'identification des endommagements basée sur les vibrations (VBDI) est un développement relativement récent dans la gestion des structures SHM, les caractéristiques physiques de la structure telles que la masse, la rigidité et les conditions aux limites affectent les caractéristiques vibratoires telles que les fréquences, les formes de mode et l'amortissement, ces changements dans les paramètres modaux sont généralement utilisés dans les techniques

VBDI tels que les changements de fréquence, de forme de mode, de courbure de forme de mode, de matrice de flexibilité et d'énergie de déformation modale. Ces changements sont obtenus à partir de la surveillance des réponses structurelles avant et après endommagements.

En utilisant ces méthodes de détection, les endommagements peuvent être détectés mais cela demande plus de temps. Le temps nécessaire pour évaluer le niveau d'endommagement peut être optimisé en améliorant la précision et l'efficacité des techniques de calcul en utilisant des algorithmes d'optimisation qui consiste en la recherche du minimum (ou du maximum) d'une certaine quantité, appelée cout ou objectif pour identifier la présence, l'emplacement et de la taille des endommagements dans une structure.

Dans cette perspective , à travers ce mémoire nous allons détecter, localiser et quantifier les endommagements dans deux structures treillis 2D et 3D .en utilisant la puissante méthode de calcul numérique « Méthode des éléments finis » à l'aide du logiciel Matlab, qui permet de calculer les paramètres physique des deux structures la masse et la rigidité afin calculer les paramètres modaux tels que les fréquences, vecteurs et valeurs propres, ces paramètres sont ensuite utiliser pour détecter les endommagements dans ces structures et finalement pour localiser et quantifier les éléments endommagés nous avons opté pour une optimisation là où nous avons utilisé deux algorithmes d'optimisation et nous avons mené une étude comparative entre ces deux algorithmes. Pour marquer ce but, nous avons établi un plan qui se compose de 4 parties, dont la première partie présente les généralités sur la méthode des éléments finis, ainsi l'application de cette méthode sur les deux structures treillis.

Dans le deuxième temps, nous expliquerons les différentes méthodes de détection des endommagements basés sur les mesures vibratoires.

Ensuite dans la 3^{ème} partie, nous présentons et nous analysons les résultats de l'application de la méthode basé sur les fréquences locales sur les deux structures.

Finalement dans la dernière partie, le but c'est d'identifier et de quantifier les endommagements obtenues dans le 3^{ème} chapitre et pour cela nous présentons nos deux algorithmes d'optimisations que nous appliquerons sur les deux treillis et ensuite nous analysons les résultats obtenus.

Et nous allons terminer ce mémoire avec une conclusion qui résume toutes les remarques et les résultats obtenus

Chapitre I. Méthode des éléments finis et dynamique des structures

I.1 Introduction :

Un corps subi souvent des changements dans la nature ce qu'on appelle les phénomènes physiques. On décrit ces phénomènes par des équations différentielles.

En mathématiques, une équation différentielle est une équation dont nous avons des inconnues qui sont des fonctions, cette équation se présente sous forme d'une relation entre ces fonctions inconnues et leurs dérivées successives. En mécanique, ces équations différentielles sont définies par le système masse-ressort à un degré de liberté. Il contient une masse accrochée à un ressort qui se déplace dans une seule direction. Son mouvement est appelé le mouvement vibratoire ou bien vibration périodique. Nous parlons d'une « vibration périodique » quand il y a un mouvement des points d'un système déformable écartés de leur position d'équilibre et qu'il revient régulièrement dans le temps qui veut dire que le système est en oscillation.

I.2 Méthodes des éléments finis :

Le système masse-ressort est souvent utilisé pour décrire le comportement physique ou dynamique des mécanismes c'est-à-dire étudier les équations différentielles ordinaires ou partielles. Pour résoudre numériquement ces équations nous appliquerons la **méthode des éléments finis**.

Chaque système d'équations linéaires peut se représenter par une matrice, ces matrices sont ensuite assemblées ce qui forme une grande matrice ; la résolution de ce système global donne la solution approchée au problème. ¹

La méthode des éléments finis est désormais la plus forte méthode numérique possible

que l'on puisse appliquer et qui sert à analyser n'importe quelle structure technique complexe dans le domaine de l'ingénierie informatique. Bien que la méthode des éléments finis ait été initialement élaborée pour résoudre des problèmes structuraux. Grâce à de nombreuses recherches approfondies. Il est désormais possible de résoudre divers problèmes de terrain tels que :

- Les champs thermiques
- Les champs d'écoulement.
- Les champs électromagnétiques.

Cette méthode est maintenant utilisée à grande échelle dans les domaines de l'ingénierie civile, hydraulique, mécanique et aéronautique.

La méthode des éléments finis peut être résumée comme suit :

- Décomposer le milieu original en un nombre fini d'éléments triangulaires en utilisant des lignes droites virtuelles. Ces lignes sont les limites des éléments dont l'intersection de plusieurs lignes droites est nommée nœud.
- En admettant que les éléments sont interconnectés sur les nœuds et que les déplacements nodaux sont considérés comme les variables inconnues de base.
- Donner une fonction qui permettra de représenter de façon spécifique le déplacement de tout point de l'élément par les déplacements de 3 nœuds ; nous appellerons cette fonction "fonction de déplacement".
- Par le biais d'une fonction de déplacement, on utilise le déplacement nodal pour représenter de manière précise la déformation de n'importe quel point de l'élément ; ensuite, par la loi de Hooke généralisée, on utilise le déplacement nodal pour représenter de manière précise la contrainte de tout point de l'élément.
- Dériver par les principes de l'énergie, la force nodale équivalente à l'état de contrainte interne de l'élément ; puis par la relation entre la contrainte de l'élément et le déplacement nodal nous allons établir la relation entre la force nodale équivalente et le déplacement nodal.

Cette étape est la plus importante pour résoudre les problèmes de contrainte par

la méthode des éléments finis.

- Répartir les charges de chaque élément entre les nœuds selon le principe d'équivalence statique.
- Représenter l'équation d'équilibre statique par des déplacements nodaux et ensuite l'établir sur chaque nœud pour obtenir un système linéaire d'équations ; la solution de cet ensemble d'équations donnera les déplacements nodaux. Nous obtiendrons de même la contrainte de chaque élément.

Nous pouvons aussi prendre la force interne de chaque élément comme la variable inconnue de base pour la résolution des problèmes (calculer les matrices), ce que l'on appelle la "méthode des forces". En revanche, la méthode des forces est plus compliquée par rapport à la méthode des déplacements pour le calcul de la matrice. Elle est rarement appliquée dans le travail réel, contrairement à la méthode de déplacement qui est largement appliquée dans le calcul pratique des structures.

I.2.1 Application de la méthode des éléments finis sur élément barre :

Nous définirons une barre comme étant un élément de structure rectiligne dont les dimensions transversales sont faibles par rapport à la longueur et qui ne travaille qu'en traction-compression le long de son axe.²

I.2.1.1 Barre a 1 D :(déplacement selon l'axe x)

Pour la barre a 1 D est une barre droite horizontale **i,j** qui possède un déplacement horizontal à chaque extrémité de l'élément, c'est-à-dire u_i et u_j . L'élément possède donc deux degrés de liberté, et la matrice de rigidité doit être sous une forme du second ordre. La condition de contrainte de l'élément peut être analysée dans deux états :

- **Condition 1 :**

$u_i = u_i ; u_j = 0$. À ce moment, le nœud **j** est fixé (pas de déplacement).

Déformation de l'élément :

$$\varepsilon = -\frac{u_j}{l} \quad (1.1)$$

Contrainte élémentaire :

$$\sigma = E\varepsilon = -\frac{Eu_j}{l} \quad (1.2)$$

Dans la théorie de la résistance des matériaux, Nous considérons la contrainte de traction comme une force positive. Cependant, dans la méthode des éléments finis, la force nodale vers la droite est considérée comme positive(même sens que le déplacement u_j)

Par conséquent, Nous ajoutons un signe négatif pour la force nodale du nœud i (gauche) pour obtenir une contrainte positive :

Force nodale à l'extrémité gauche de l'élément :

$$U_i = -A\sigma = \frac{AE}{l} u_j \quad (1.3)$$

Force nodale à l'extrémité droite de l'élément :

$$U_j = A\sigma = -\frac{AE}{l} u_i \quad (1.4)$$

- **Condition 2 :**

$u_i = 0$; $u_j = u_j$. (Cette condition est l'opposée de la 1^{er} condition) . À ce moment, c'est le nœud i qui est fixé (pas de déplacement).

Déformation de l'élément :

$$\varepsilon = -\frac{u_j}{l} \quad (1.5)$$

Contrainte élémentaire :

$$\sigma = \frac{Eu_j}{l} \quad (1.6)$$

La contrainte reste toujours positive au nœud **j** (droit).

N associant les résultats des deux conditions précédents, les forces nodales de l'élément sont donnés comme suit :

$$U_i = \frac{AE}{l} u_i - \frac{AE}{l} u_j \quad (1.7)$$

$$U_j = -\frac{AE}{l} u_i + \frac{AE}{l} u_j \quad (1.8)$$

sous forme matricielle :

$$\begin{Bmatrix} U_i \\ U_j \end{Bmatrix} = \frac{AE}{l} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_i \\ u_j \end{Bmatrix} = [k]^e \begin{Bmatrix} u_i \\ u_j \end{Bmatrix} \quad (1.9)$$

d'où nous allons prendre $[k]^e$ comme matrice de rigidité de l'élément

$$[k]^e = \frac{AE}{l} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (1.10)$$

La force axiale (selon l'axe x) de l'élément peut être formulée sous la forme suivante :

$$N = \frac{AE}{l} \begin{bmatrix} -1 & 1 \end{bmatrix} \begin{Bmatrix} u_i \\ u_j \end{Bmatrix} = [S] \begin{Bmatrix} u_i \\ u_j \end{Bmatrix} \quad (1.11)$$

D'où, la force axiale est considérée comme une contrainte généralisée. Par conséquent, la matrice $[S]$ est également appelée matrice de contrainte de l'élément, s'écrit comme suit :

$$[S] = \frac{AE}{l} \begin{bmatrix} -1 & 1 \end{bmatrix} \quad (1.12)$$

En revanche, en plus des déplacements horizontaux qui se produisent aux nœuds **i** et **j**, il peut se produire aussi des déplacements verticaux mais dans des conditions de faible déformation, ces derniers n'ont aucune influence sur la force interne des barres donc on les néglige.

Nous pouvons élargir la matrice de rigidité de l'élément de barre **1D** sous une forme du quatrième ordre. Par conséquent, la force nodale de l'élément est la suivante :

$$\begin{Bmatrix} U_i \\ V_i \\ U_j \\ V_j \end{Bmatrix} = \frac{AE}{l} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \\ u_j \\ v_j \end{Bmatrix} \quad (1.13)$$

Dans cette formule, le déplacement nodal vertical v_i et v_j ainsi que la force nodale verticale V_i et V_j sont introduits en ajoutant des zéros à la matrice de rigidité

La formule générale pour la force axiale de l'élément de barre horizontale est la suivante :

La force axiale de l'élément :

$$N = \frac{AE}{l} \begin{bmatrix} -1 & 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \\ u_j \\ v_j \end{Bmatrix} = [S] \{\delta\}^e \quad (1.14)$$

D'où $\{\delta\}^e = [u_i \ v_i \ u_j \ v_j]^T$ chaque élément possède six déplacements nodaux .

I.2.1.2 Barre a 2 D : (déplacement selon les axes (x,y))

Pour un élément de barre inclinée tel qu'illustré à la Figure 1,

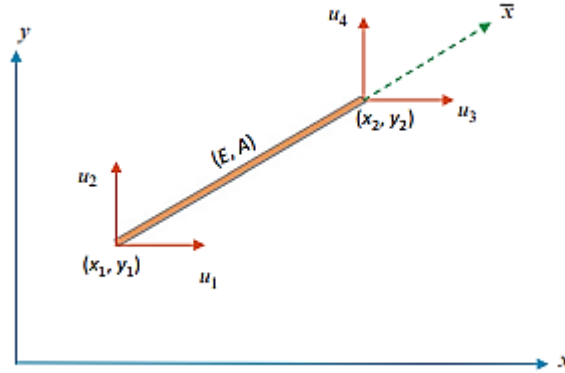


Figure 1. Élément Barre 2 dimensions

Le déplacement nodal au nœud i est le suivant :

$$\{\delta_i\} = \begin{Bmatrix} u_i \\ v_i \end{Bmatrix} \quad (1.15)$$

dans lequel u_i et v_i sont, respectivement, les composantes horizontale et verticale du déplacement nodal.

La force nodale au nœud i correspond à :

$$\{F_i\} = \begin{Bmatrix} U_i \\ V_i \end{Bmatrix} \quad (1.16)$$

où U_i et V_i sont, respectivement, les composantes horizontale et verticale de la force nodale.

Les symboles du déplacement nodal u_i et v_i de la force nodale $\{F_i\}$ doivent être positifs lorsque la direction est la même que celle des axes de coordonnées x. ou négatifs dans le cas contraire.

Tout d'abord, nous allons faire une analyse de la relation **déformation-déplacement** de la barre inclinée puis définir la longueur de l'élément barre qu'on note l, et que nous dérivons de la relation géométrique :

$$l^2 = (x_j - x_i)^2 + (y_j - y_i)^2 \quad (1.17)$$

En prenant la dérivée différentielle des deux côtés de la formule précédente, nous arrivons au résultat suivant :

$$l dl = (x_j - x_i)(dx_j - dx_i) + (y_j - y_i)(dy_j - dy_i) \quad (1.18)$$

D'où :

$$a = \cos\theta = \frac{x_j - x_i}{l}, \beta = \sin\theta = \frac{y_j - y_i}{l} \quad (1.19)$$

Après que l'élément se déforme sous la contrainte, les coordonnées du nœud i passeront de (x_i, y_i) en $(x_i + u_i, y_i + v_i)$, d'où :

$$dx_i = u_i, dy_i = v_i \quad (1.20)$$

De la même manière,

$$dx_j = u_j, dy_j = v_j \quad (1.21)$$

En substituant les formules (1.20) et (1.21) précédente dans la formule (1.18) et en la divisant par l , nous obtiendrons la formule de la déformation de l'élément barre comme suit :

$$\varepsilon = \frac{dl}{l} = \frac{a}{l}(-u_i + u_j) + \frac{\beta}{l}(-v_i + v_j) \quad (1.22)$$

La force axiale de la barre inclinée est sous cette forme :

$$N = AE\varepsilon = \frac{AE}{l} [a(-u_i + u_j) + \beta(-v_i + v_j)] \quad (1.23)$$

La force de traction est toujours positive pour la force axiale N.

La valeur absolue de la force nodale est égale aux composantes horizontale et verticale de la force axiale N, donc :

$$U_i = -N \cos \theta \quad V_i = -N \sin \theta \quad (1.24)$$

$$U_j = N \cos \theta, V_j = N \sin \theta \quad (1.25)$$

En remplaçant les formules (1.24) et (1.25) dans la formule (1.23), nous obtiendrons la force nodale de l'élément de barre inclinée que nous pourrions représenter sous la forme matricielle comme suit :

$$\{F\}^e = \begin{Bmatrix} U_i \\ V_i \\ U_j \\ V_j \end{Bmatrix} = \frac{AE}{l} \begin{bmatrix} a^2 & a\beta & -a^2 & -a\beta \\ a\beta & \beta^2 & -a\beta & -\beta^2 \\ -a^2 & -a\beta & a^2 & a\beta \\ -a\beta & -\beta^2 & a\beta & \beta^2 \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \\ u_j \\ v_j \end{Bmatrix} = [k]^e \{\delta\}^e \quad (1.26)$$

Avec $[k]^e$ qui représente la matrice de rigidité de l'élément de barre inclinée elle est symétrique. Nous pourrions également exprimer la formule

(1.26) par une matrice divisée comme suit :

$$\{F\}^e = \begin{Bmatrix} F_i \\ F_j \end{Bmatrix} = \begin{bmatrix} k_{ii} & k_{ij} \\ k_{ji} & k_{jj} \end{bmatrix} \begin{Bmatrix} \delta_i \\ \delta_j \end{Bmatrix} \quad (1.27)$$

Avec :

$$\{F_i\} = \begin{Bmatrix} U_i \\ V_i \end{Bmatrix}, \{F_j\} = \begin{Bmatrix} U_j \\ V_j \end{Bmatrix}, \{\delta_i\} = \begin{Bmatrix} u_i \\ v_i \end{Bmatrix}, \{\delta_j\} = \begin{Bmatrix} u_j \\ v_j \end{Bmatrix} \quad (1.28)$$

F_i et F_j sont les forces nodales aux extrémités de l'élément .

U_i, u_j et V_j, v_j sont respectivement les composantes nodales des nœuds i, j .

Et nous notons aussi les sous matrices de la formule de la force nodale élémentaire comme décrit ci -dessous :

$$[k_{ii}] = [k_{jj}] = \frac{AE}{l} \begin{bmatrix} a^2 & a\beta \\ a\beta & \beta^2 \end{bmatrix}, [k_{ij}] = [k_{ji}] = \frac{AE}{l} \begin{bmatrix} -a^2 & -a\beta \\ -a\beta & -\beta^2 \end{bmatrix}, \quad (1.29)$$

Dont :

$$a = \cos\theta, \beta = \sin\theta \quad (1.30)$$

I.2.1.3 Barre a 3 D : (déplacement selon les axes (x,y,z))

Nous allons considérer maintenant une barre a 2 nœuds dans le système **3D** à coordonnées **(x, y, z)** Figure 2 . En suivant une procédure similaire à celle de **2D** nous allons obtenir la matrice de rigidité de cet élément

Il est bien évident que tout ensemble de déplacements d_{ix}, d_{iy}, d_{iz} d'un nœud i dans le système **x-y-z** provoque un déplacement d_{ix} le long de l'axe **x** donné ainsi :

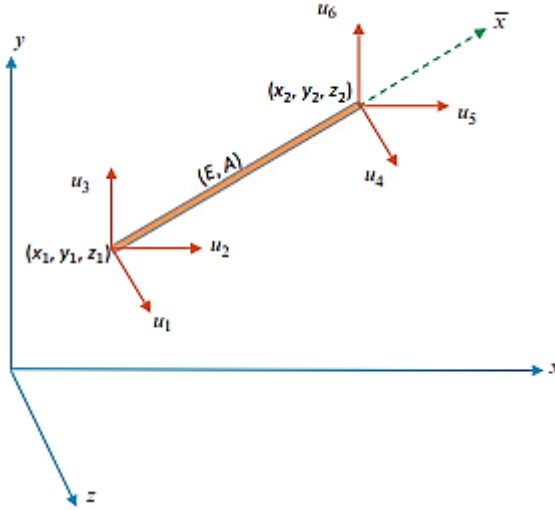


Figure 2. Élément Barre a 3 dimensions

$$d_{ix} = (\cos\theta_x)d_{ix} + (\cos\theta_y)d_{iy} + (\cos\theta_z)d_{iz} \quad (1.31)$$

Avec :

$$\cos\theta_x = \frac{x_2 - x_1}{L}, \quad \cos\theta_y = \frac{y_2 - y_1}{L}, \quad \cos\theta_z = \frac{z_2 - z_1}{L} \quad (1.32)$$

Dans les équations précédentes, (x_1, y_1, z_1) et (x_2, y_2, z_2) sont les coordonnées des nœuds 1 et 2 de la barre, et L est la longueur de la barre qui est donnée par :

$$L = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (1.33)$$

L'application de l'équation (1.31) aux nœuds 1 et 2 aboutit aux résultats suivants :

$$d_{1x} = C_x d_{1x} + C_y d_{1y} + C_z d_{1z} \quad (1.34)$$

$$d_{2x} = C_x d_{2x} + C_y d_{2y} + C_z d_{2z}$$

Dont :

$$C_x = \cos\theta_x, C_y = \cos\theta_y, C_z = \cos\theta_z \quad (1.35)$$

Nous pourrions écrire les équations (1.34) sous la forme matricielle suivante :

$$\begin{Bmatrix} d_{1x} \\ d_{2x} \end{Bmatrix} = \begin{bmatrix} C_x & C_y & C_z & 0 & 0 & 0 \\ 0 & 0 & 0 & C_x & C_y & C_z \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{1z} \\ d_{2x} \\ d_{2y} \\ d_{2z} \end{Bmatrix} \quad (1.36)$$

De la même manière, la résultante r_{ix} d'un ensemble de forces nodales (r_{ix} , r_{iy} , r_{iz}) appliquées à un nœud i peut être exprimée par l'équation ci-après :

$$\begin{Bmatrix} r_{1x} \\ r_{2x} \end{Bmatrix} = \begin{bmatrix} C_x & C_y & C_z & 0 & 0 & 0 \\ 0 & 0 & 0 & C_x & C_y & C_z \end{bmatrix} \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{1z} \\ r_{2x} \\ r_{2y} \\ r_{2z} \end{Bmatrix} \quad (1.37)$$

En prenant en considération les équations

(1.36) et

(1.37), nous allons les remplacer dans l'équation $\begin{Bmatrix} r_{1x} \\ r_{2x} \end{Bmatrix} = \begin{bmatrix} EA/L & -EA/L \\ -EA/L & EA/L \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{2x} \end{Bmatrix}$ pour obtenir l'équation suivante:

$$\begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{1z} \\ r_{2x} \\ r_{2y} \\ r_{2z} \end{Bmatrix} = \frac{AE}{L} \begin{bmatrix} C_x^2 & C_x C_y & C_x C_z & -C_x^2 & -C_x C_y & -C_x C_z \\ & C_y^2 & C_y C_z & -C_x C_y & -C_y^2 & -C_y C_z \\ & & C_z^2 & -C_x C_z & -C_y C_z & -C_z^2 \\ & & & C_x^2 & C_x C_y & C_x C_z \\ & & & & C_y^2 & C_y C_z \\ & & & & & C_z^2 \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{1z} \\ d_{2x} \\ d_{2y} \\ d_{2z} \end{Bmatrix} \quad (1.38)$$

symetrique

I.2.2 L'assemblage des matrices :

Pour une structure en treillis, les matrices de rigidité structurelle, K^S est déterminée comme la somme de toutes les matrices de rigidité des barres correspondantes K_i^S . Il est essentiel que la matrice de rigidité de l'élément soit représentée dans un système de coordonnées global avec tous les termes de déplacement et de force nodaux de la structure représentés dans une équation de matrice force-déplacement globale.¹

Pour assembler la matrice de rigidité structurelle des systèmes de coordonnées globales, il est habituellement effectué en additionnant les matrices de rigidité des membres individuels conformément à l'équation suivante :

$$K^S = \sum_{i=1}^n K_i^S \quad (1.39)$$

Où K^S est la matrice de rigidité du système de coordonnées global (structurel), et K_i^S est la matrice de rigidité de la barre i dans le système de coordonnées global (élément).

I.2.3 Les conditions aux limites :

Pour fixer la structure au sol, des conditions d'appui sont introduites de telle manière à assurer la stabilité de l'ensemble. Suivant la nature des degrés de liberté bloqués, ces supports peuvent reconstituer des appuis ponctuels, des appuis rotules ou des encastremements mais également correspondre à des déplacements nuls ou imposés.

I.2.4 Transformation des coordonnées :

La matrice de rigidité, et la matrice de masse, ont été dérivées en référence aux coordonnées nodales associées au système de coordonnées local, il est nécessaire de transformer ces matrices en un système de référence commun, le système de coordonnées

global.

En effectuant une rotation des coordonnées, en supprimant les coordonnées angulaires dans l'équation et en renommant les coordonnées restantes, nous obtiendrons la transformation suivante pour les forces nodales :

$$\{F\} = [T]\{\bar{F}\} \quad (1.40)$$

Sous forme matricielle :

$$\begin{Bmatrix} U_i \\ V_i \\ U_j \\ V_j \end{Bmatrix} = \begin{bmatrix} a & \beta & 0 & 0 \\ -\beta & a & 0 & 0 \\ 0 & 0 & a & \beta \\ 0 & 0 & -\beta & a \end{bmatrix} \begin{Bmatrix} \bar{U}_i \\ \bar{V}_i \\ \bar{U}_j \\ \bar{V}_j \end{Bmatrix} \quad (1.41)$$

Ou

$$[T] = \begin{bmatrix} a & \beta & 0 & 0 \\ -\beta & a & 0 & 0 \\ 0 & 0 & a & \beta \\ 0 & 0 & -\beta & a \end{bmatrix} \quad (1.42)$$

Dont [T] est la matrice de transformation (matrice de passage) .

Pour le déplacement nodal, il existe également une relation similaire entre la coordonnée locale et la coordonnée globale :

$$\{\delta\} = [T]\{\bar{\delta}\} \quad (1.43)$$

➤ Matrice de rigidité :

Maintenant, nous allons transférer la matrice de rigidité de l'élément barre dans le système de coordonnées local dans la matrice de rigidité du système de coordonnées global. Sachant que :

$$\{F\} = [k]^e \{\delta\} \quad (1.44)$$

En remplaçant (1.40) nous obtiendrons :

$$[T]\{\bar{F}\} = [\bar{k}]^e [T] \{\delta\} \quad (1.45)$$

En multipliant $[T]^{-1}$ des deux côtés de la formule précédente, nous avons :

$$\{\bar{F}\} = [T]^{-1} [\bar{k}]^e [T] \{\delta\} \quad (1.46)$$

Comme $[T]$ est une matrice orthogonale $[T]^{-1} = [T]^T$, donc :

$$\{\bar{F}\} = [T]^T [\bar{k}]^e [T] \{\delta\} \quad (1.47)$$

Avec :

$$\{\bar{F}\} = [k]^e \{\delta\} \quad (1.48)$$

Donc :

$$[k]^e = [T]^T [\bar{k}]^e [T] \quad (1.49)$$

➤ **Matrice de Masse :**

Nous remplaçons les équations (1.43) et (1.44) dans cette équation :

$$\{F\} = [\bar{M}]\{\delta\} \quad (1.50)$$

On obtient :

$$\{\bar{F}\}=[T]^T[\bar{M}][T]\{\delta\} \quad (1.51)$$

Avec $\{\bar{F}\}=[M]\{\delta\}$

L'équation devienne :

$$[M]\{\delta\}=[T]^T[\bar{M}][T]\{\delta\} \quad (1.52)$$

Nous obtiendrons l'équation de la matrice de masse comme suit :

$$[M]=[T]^T[\bar{M}][T] \quad (1.53)$$

I.3 L'analyse dynamique des structures

L'étude des vibrations porte sur les mouvements oscillatoires des corps suite à l'action des forces qui leur sont imposées. Nous pouvons dire que tout corps doté d'une masse et d'une élasticité est susceptible d'être soumis à des vibrations. Presque toute nouvelle construction mécanique est soumise à une étude intense à la vibration soit pendant la conception ou tout au long de son développement.³

I.3.1 Equation de mouvement :

Après la discrétisation des structures, les équations d'équilibre dynamique de chaque nœud dans l'état de mouvement sont les suivantes :

$$\{F_i\}+\{F_d\}+\{P(t)\}=\{F_e\} \quad (1.51)$$

où $\{F_i\}$, $\{F_d\}$, $\{P(t)\}$ et $\{F_e\}$ sont respectivement la force d'inertie, la force d'amortissement, les charges dynamiques et la force élastique, qui sont toutes des vecteurs.

Le vecteur de la force élastique peut être exprimé par le déplacement nodal $\{\delta\}$ et la

matrice de rigidité $[K]$, comme suit :

$$\{F_e\} = [K] \{\delta\} \quad (1.52)$$

Ici, l'élément K_{ij} de la matrice de rigidité $[K]$ est la force élastique au nœud i causée par le déplacement unitaire du nœud j .

Selon le principe de d'Alembert, la force d'inertie peut être exprimée par la matrice de masse $[M]$ et l'accélération du nœud $\frac{\partial^2 \{\delta\}}{\partial t^2}$ comme suit :

$$\{F_i\} = -[M] \frac{\partial^2 \{\delta\}}{\partial t^2} \quad (1.53)$$

La structure est supposée avoir un amortissement visqueux ; la force d'amortissement peut être exprimée par la matrice d'amortissement $[C]$ et la vitesse des nœuds. $\frac{\partial \{\delta\}}{\partial t}$ comme suit :

$$\{F_d\} = -[C] \frac{\partial \{\delta\}}{\partial t} \quad (1.54)$$

Ici, l'élément C_{ij} de la matrice d'amortissement est la force d'amortissement au nœud i causée par la vitesse unitaire du nœud j

En substituant les équations (1.52) et (1.53) , (1.54), dans (1.51), nous obtenons l'équation du mouvement comme suit :

$$[M] \frac{\partial^2 \{\delta\}}{\partial t^2} + [C] \frac{\partial \{\delta\}}{\partial t} + [K] \{\delta\} = \{P(t)\} \quad (1.55)$$

On prend :

$$\{\dot{\delta}\} = \frac{\partial \{\delta\}}{\partial t}, \{\ddot{\delta}\} = \frac{\partial^2 \{\delta\}}{\partial t^2} \quad (1.56)$$

L'équation du mouvement devienne comme suit :

$$[M]\{\ddot{\delta}\} + [C]\{\dot{\delta}\} + [K]\{\delta\} = \{P(t)\} \quad (1.57)$$

Selon le principe de la dynamique la somme des forces extérieurs (forces dynamiques) appliqués à un solide est égale au produit de la masse du solide par l'accélération de son centre d'inertie ce qui nous donne :

$$P(t) = - M(a) \quad (1.58)$$

Donc l'équation du mouvement pour un système à 1 DDL amortis est comme suit :

$$[M]\{\ddot{\delta}\} + [C]\{\dot{\delta}\} + [K]\{\delta\} = \{- M(a)\} \quad (1.59)$$

I.3.2 Système Masse –Ressort (non amorti):

Le système constitué d'une masse et d'un ressort, en mouvement libre, est le plus simple des systèmes vibrants. Malgré tout, il permet d'expliquer l'essentiel des phénomènes que l'on rencontre dans la réalité. C'est pourquoi, nous allons établir les équations de son mouvement pour cela, nous écrivons l'équilibre des forces intérieurs et extérieurs au système.

Equation du mouvement :

En pose $P(t)=0$ pour un système libre dans l'équation (1.57), nous obtiendrons l'équation de vibration libre. Dans la pratique, l'amortissement a peu d'effet sur la fréquence naturelle et le mode de vibration, on obtient donc les équations de vibration libre sans amortissement en négligeant la force d'amortissement :

$$[K]\{\delta\} + [M]\{\ddot{\delta}\} = 0 \quad (1.60)$$

Supposons que la structure ait un mouvement harmonique simple comme suit :

$$\{\delta\} = \{\phi\} \cos \omega t \quad (1.61)$$

En substituant l'équation précédente dans l'équation (1.60), l'équation homogène est obtenue comme suit :

$$[K]\{\phi\} = \omega^2 [M] \{\phi\} \quad (1.62)$$

Ou sous une autre forme :

$$([K]\{\phi\} - \omega^2 [M])\{\phi\} = 0 \quad (1.63)$$

En vibration libre, les amplitudes $\{\phi\}$ de chaque nœud de la structure ne sont pas toutes nulles, par conséquent le déterminant de la matrice entre parenthèses de l'équation (1.63) doit être égal à zéro. Et nous obtiendrons l'équation de fréquence naturelle de la structure comme suit :

$$|[K] - \omega^2 [M]| = 0 \quad (1.64)$$

I.3.3 Matrice de Masse :

Selon le principe de d'Alembert, la force d'inertie qui agit sur le volume élémentaire d'une partie minuscule d'une structure en mouvement est :

$$\{p_i\} = -\rho \frac{\partial^2 \{r\}}{\partial t^2} \quad (1.65)$$

Ici ρ est la densité du matériau. Après la discrétisation des structures, on considère un

élément de la structure, et nous prenons la forme suivante de la fonction de déplacement :

$$\{r\} = [N]\{\delta\}^e \quad (1.66)$$

En substituant l'équation précédente dans l'équation (1.65), nous obtenons :

$$\{p_i\} = -\rho[N] \frac{\partial^2 \{\delta\}^e}{\partial t^2} \quad (1.67)$$

Ensuite, en utilisant la formule de transposition de la charge, nous pouvons calculer la force d'inertie exercée sur le nœud de l'élément comme suit :

$$\{F_i\}^e = \int [N]^T \{p_i\} dV = - \int [N]^T \rho [N] dV \frac{\partial^2 \{\delta\}^e}{\partial t^2} \quad (1.68)$$

Nous prenons :

$$\{F_i\}^e = -[m] \{\ddot{\delta}\}^e \quad (1.69)$$

Il est clair que la matrice de masse de l'élément est :

$$[m] = \int [N]^T \rho [N] dV \quad (1.70)$$

I.3.4 Matrice de Masse pour une barre 1D :

Selon le principe D'Alembert., la distribution de la force d'inertie dans le corps peut être utilisée pour dériver la matrice de masse $[m]$ en utilisant les fonctions de forme $N_1(x)$ et $N_2(x)$:

$$[m] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix} = \int \begin{Bmatrix} N_1(x) \\ N_2(x) \end{Bmatrix} \int \rho \ddot{d}(x) dV \quad (1.71)$$

Nous allons prendre :

$$\ddot{d}(x) = [N_1(x) \ N_2(x)] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix} \quad (1.72)$$

L'équation (1.71) devienne :

$$[m] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix} = \rho \int \begin{Bmatrix} N_1(x) \\ N_2(x) \end{Bmatrix} [N_1(x) \ N_2(x)] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix} dV \quad (1.73)$$

Puisque les accélérations nodales $\begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix}$ sont indépendantes de la variable x , l'équation ci-dessus peut s'écrire comme suit :

$$[m] = \rho \int \begin{Bmatrix} N_1(x) \\ N_2(x) \end{Bmatrix} [N_1(x) \ N_2(x)] dV \quad (1.74)$$

Pour les barres à section uniforme, la valeur de l'élément de volume est de :

$$dV = A dx \quad (1.75)$$

On décrit les fonctions de forme comme suit :

$$N_1(x) = 1 - \frac{x}{L}, \quad N_2(x) = \frac{x}{L} \quad (1.76)$$

En prenant en compte les équations (1.75) et (1.76), l'équation (1.74) donne :

$$[m] = \rho A \int_0^L \begin{Bmatrix} \left(1 - \frac{x}{L}\right) \\ \left(\frac{x}{L}\right) \end{Bmatrix} \begin{bmatrix} \left(1 - \frac{x}{L}\right) & \left(\frac{x}{L}\right) \end{bmatrix} dx \quad (1.77)$$

En développant chaque terme de la matrice et en intégrant puis en remplaçons les bornes **0** et **L**, nous obtiendrons la matrice de masse pour une barre 1 D que nous décrivons comme suit :

$$[m] = \frac{\rho AL}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad (1.78)$$

I.3.5 Matrice de masse pour une barre 2D :

Notre objectif est maintenant de transformer l'équation (système masse ressort x) par rapport au système de coordonnées global x-y on prend l'équation de l'élément barre dans les coordonnées locales :

$$[\bar{k}] \begin{Bmatrix} d_{1x} \\ d_{2x} \end{Bmatrix} + [\bar{m}] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix} = \begin{Bmatrix} r_{1x} \\ r_{2x} \end{Bmatrix} \quad (1.79)$$

On a :

$$\begin{Bmatrix} d_{1x} \\ d_{2x} \end{Bmatrix} = [\bar{T}] \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{2x} \\ d_{2y} \end{Bmatrix} \quad (1.80)$$

En dérivant deux fois ce vecteur de déplacement nous allons obtenir le vecteur d'accélération suivant :

$$\begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix} = [\bar{T}] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \end{Bmatrix} \quad (1.81)$$

Nous représentons le vecteur des déplacements de forces comme ceci :

$$\begin{Bmatrix} r_{1x} \\ r_{2x} \end{Bmatrix} = [\bar{T}] \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{2x} \\ r_{2y} \end{Bmatrix} \quad (1.82)$$

Ou la matrice de transfert (passage) est décrite comme suit :

$$[\bar{T}] = \begin{bmatrix} C & S & 0 & 0 \\ 0 & 0 & C & S \end{bmatrix} \quad (1.83)$$

Avec : $C=\cos(\theta)$, $S=\sin(\theta)$

En remplaçant tout ca dans l'équation(1.79) .l'équation de l'élément devienne :

$$[\bar{k}][\bar{T}] \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{2x} \\ d_{2y} \end{Bmatrix} + [\bar{m}][\bar{T}] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \end{Bmatrix} = \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{2x} \\ r_{2y} \end{Bmatrix} \quad (1.84)$$

Pour lier les coordonnées globales, nous devons inverser la matrice T . Cependant, cela n'est pas possible car T n'est pas une matrice carrée. Par conséquent, les équations **(1.80)** et **(1.81)** doivent être exprimées sous la forme suivante :

Déplacement : (1.85)

$$\begin{Bmatrix} d_{1x} \\ 0 \\ d_{2x} \\ 0 \end{Bmatrix} = \begin{bmatrix} C & S & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & C & S \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{2x} \\ d_{2y} \end{Bmatrix}$$

Accélération : (1.86)

$$\begin{Bmatrix} \ddot{d}_{1x} \\ 0 \\ \ddot{d}_{2x} \\ 0 \end{Bmatrix} = \begin{bmatrix} C & S & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & C & S \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \end{Bmatrix}$$

Déplacement des forces : (1.87)

$$\begin{Bmatrix} r_{1x} \\ 0 \\ r_{2x} \\ 0 \end{Bmatrix} = \begin{bmatrix} C & S & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & C & S \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{2x} \\ r_{2y} \end{Bmatrix}$$

Nous ferons de même pour l'équation de rigidité [K] :

$$\begin{Bmatrix} r_{1x} \\ 0 \\ r_{2x} \\ 0 \end{Bmatrix} = \begin{bmatrix} EA/L & 0 & -EA/L & 0 \\ 0 & 0 & 0 & 0 \\ -EA/L & 0 & EA/L & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} d_{1x} \\ 0 \\ d_{2x} \\ 0 \end{Bmatrix} \quad (1.88)$$

Suivant le même concept, le membre de l'équation de masse, suivante :

$$[m][\ddot{d}] = \frac{1}{2} \rho AL \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{2x} \end{Bmatrix} \quad (1.89)$$

Peut-être écrite sous la forme alternative suivante :

$$[m][\ddot{d}] = \frac{1}{2} \rho AL \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{d}_{1x} \\ 0 \\ \ddot{d}_{2x} \\ 0 \end{Bmatrix} \quad (1.90)$$

Par conséquent, la matrice modifiée de masse globale est créée comme suit :

$$[m^*] = \frac{1}{2} \rho AL \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.91)$$

De même, la matrice de masse modifiée est :

$$[m^*] = \frac{1}{6} \rho AL \begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.92)$$

Selon l'équation (1.88) la matrice de rigidité modifiée est :

$$[k^*] = \begin{bmatrix} EA/L & 0 & -EA/L & 0 \\ 0 & 0 & 0 & 0 \\ -EA/L & 0 & EA/L & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.93)$$

L'équation

(1.84) peut alors être exprimée sous la forme équivalente suivante :

$$[k^*] \begin{Bmatrix} d_{1x} \\ 0 \\ d_{2x} \\ 0 \end{Bmatrix} + [m^*] \begin{Bmatrix} \ddot{d}_{1x} \\ 0 \\ \ddot{d}_{2x} \\ 0 \end{Bmatrix} = \begin{Bmatrix} r_{1x} \\ 0 \\ r_{2x} \\ 0 \end{Bmatrix} \quad (1.94)$$

La fonction de transfert devienne :

$$[T] = \begin{bmatrix} C & S & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & C & S \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.95)$$

L'équation

(1.84) devienne alors comme suit :

$$[T]^T [k^*] [T] \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{2x} \\ d_{2y} \end{Bmatrix} + [T]^T [m^*] [T] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \end{Bmatrix} = \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{2x} \\ r_{2y} \end{Bmatrix} \quad (1.96)$$

Où :

$$[k] \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{2x} \\ d_{2y} \end{Bmatrix} + [m] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \end{Bmatrix} = \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{2x} \\ r_{2y} \end{Bmatrix} \quad (1.97)$$

la matrice de masse en coordonnées globales xy :

$$[m] = \frac{1}{2} \rho AL \cdot [T]^T \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} [T] \quad (1.98)$$

Et la matrice de masse en coordonnées globales xy peut s'écrire sous cette forme :

$$[m] = \frac{1}{6} \rho AL \cdot [T]^T \cdot \begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} [T] \quad (1.99)$$

I.3.6 Matrice de masse pour une barre 3D :

La dérivation de la matrice de masse d'un élément de barre tridimensionnel (3D) peut être basée sur le même concept que l'élément de barre 2D.

Pour l'élément 3D, l'équation de mouvement a maintenant la forme :

$$[k^*] \begin{Bmatrix} d_{1x} \\ 0 \\ 0 \\ d_{2x} \\ 0 \\ 0 \end{Bmatrix} + [m^*] \begin{Bmatrix} \ddot{d}_{1x} \\ 0 \\ 0 \\ \ddot{d}_{2x} \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} r_{1x} \\ 0 \\ 0 \\ r_{2x} \\ 0 \\ 0 \end{Bmatrix} \quad (1.100)$$

Où :

$$[k^*] = \begin{bmatrix} EA/L & 0 & 0 & -EA/L & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -EA/L & 0 & 0 & EA/L & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.101)$$

En revanche, les déplacements, les accélérations et les forces nodales peuvent être exprimées comme suit :

$$\begin{Bmatrix} d_{1x} \\ 0 \\ 0 \\ d_{2x} \\ 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} C_x & C_y & C_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_x & C_y & C_z \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{1z} \\ d_{2x} \\ d_{2y} \\ d_{2z} \end{Bmatrix} \quad (1.102)$$

$$\begin{Bmatrix} \ddot{d}_{1x} \\ 0 \\ 0 \\ \ddot{d}_{2x} \\ 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} C_x & C_y & C_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_x & C_y & C_z \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{1z} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \\ \ddot{d}_{2z} \end{Bmatrix} \quad (1.103)$$

$$\begin{Bmatrix} r_{1x} \\ 0 \\ 0 \\ r_{2x} \\ 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} C_x & C_y & C_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_x & C_y & C_z \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{1z} \\ r_{2x} \\ r_{2y} \\ r_{2z} \end{Bmatrix} \quad (1.104)$$

Avec : $C_x = \cos\theta_x$, $C_y = \cos\theta_y$, $C_z = \cos\theta_z$

En exprimant la matrice de passage comme suit :

$$[T] = \begin{bmatrix} C_x & C_y & C_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & C_x & C_y & C_z \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.105)$$

Nous remplaçons tout dans l'équation

(1.100) :

$$[k^*][T] \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{1z} \\ d_{2x} \\ d_{2y} \\ d_{2z} \end{Bmatrix} + [m^*][T] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{1z} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \\ \ddot{d}_{2z} \end{Bmatrix} = [T] \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{1z} \\ r_{2x} \\ r_{2y} \\ r_{2z} \end{Bmatrix} \quad (1.106)$$

Par conséquent, en multipliant les deux côtés de l'équation ci-dessus par la transposée de $[T]$ nous obtiendrons la formule suivante :

$$[k] \begin{Bmatrix} d_{1x} \\ d_{1y} \\ d_{1z} \\ d_{2x} \\ d_{2y} \\ d_{2z} \end{Bmatrix} + [m] \begin{Bmatrix} \ddot{d}_{1x} \\ \ddot{d}_{1y} \\ \ddot{d}_{1z} \\ \ddot{d}_{2x} \\ \ddot{d}_{2y} \\ \ddot{d}_{2z} \end{Bmatrix} = \begin{Bmatrix} r_{1x} \\ r_{1y} \\ r_{1z} \\ r_{2x} \\ r_{2y} \\ r_{2z} \end{Bmatrix} \quad (1.107)$$

Matrice de masse en coordonnées globales xyz :

$$[m] = \frac{1}{2} \rho AL \cdot [T]^T \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} [T] \quad (1.108)$$

Et elle peut aussi s'écrire sous cette forme :

$$[m] = \frac{1}{6} \rho AL \cdot [T]^T \cdot \begin{bmatrix} 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} [T] \quad (1.109)$$

I.4 Programmation sur Matlab :

Les fonctions pour les applications d'éléments finis sont toutes des fonctions MATLAB (fichiers .m), comme décrit dans le manuel MATLAB. L'analyse par éléments finis peut être effectuée de manière interactive ou par lots ou en mode batch.

En mode interactif, les fonctions sont évaluées une par une dans la fenêtre de commande MATLAB dans la fenêtre de commande de MATLAB. En mode batch, une séquence de fonctions est écrite dans un fichier nommé **.m-file**, et évaluée en écrivant le nom du fichier dans la fenêtre de commande. Le mode batch est un moyen plus souple d'effectuer une analyse par éléments finis car le fichier .m peut être écrit dans n'importe quelle langue. et pour cela nous avons utilisé le programme informatique CALFEM qui est une boîte à outils MATLAB pour les applications d'éléments finis. Il concerne principalement les fonctions d'éléments finis, mais il contient également des descriptions de certaines fonctions MATLAB souvent utilisées.⁴

I.4.1 Barre 1 D et 2D :

➤ **Syntaxe :**

$[Ke, Me] = \text{bar2e2D}(Ex, Ey, ep)$

➤ **Description :**

bar2e2D fournit la matrice de rigidité globale **Ke** et la matrice de masse **Me** pour un élément de barre à deux dimensions.

Les variables d'entrée

ex= [x1 x2]

ey=[y1 y2]

ep=[E A Rho*A] fournir les coordonnées nodales de l'élément x1, y1, x2 et y2, le module d'élasticité E, et l'aire de la section transversale A et la masse volumique **Rho** .

➤ **Théorie :**

La matrice de rigidité de l'élément $[k]^e$ est calculée selon la formule suivante :

$$[k]^e = [T]^T [\bar{k}]^e [T] \quad (1.110)$$

Avec :

$$[k]^e = \frac{AE}{l} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (1.111)$$

Nous écrivons la matrice de Masse de l'élément sur la même forme que celle de rigidité :

$$[M] = [T]^T [\bar{M}] [T] \quad (1.112)$$

Avec :

$$[\bar{M}] = \frac{\rho AL}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad (1.113)$$

La matrice de transformation T s'écrit sous cette forme :

$$[\bar{T}] = \begin{bmatrix} C & S & 0 & 0 \\ 0 & 0 & C & S \end{bmatrix} \quad (1.114)$$

Elle contient les cosinus et sinus de direction : $C = \cos \theta = \frac{x_j - x_i}{L}$; $S = \sin \theta = \frac{y_j - y_i}{L}$

Où la longueur est égale à :

$$L = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \quad (1.115)$$

➤ Programmation :

```
function [Ke,Me]=bar2e2D(Ex,Ey,ep)
%
%-----
% PURPOSE
% calculer la matrice de rigidité et de masse pour une barre a 2
dimensions .
%
% INPUT: Ex = [x1 x2];
%         Ey = [y1 y2];      variables d'entrée (element node coordinates)
%
%         ep = [E A m]      E: Module de Young
%                           A: l'aire de la section transversale
% %
%                           m=rho*A
%                           Rho: masse volumique
%
% OUTPUT: Ke :matrice de rigidité , dim(Ke)= 4 x 4
%-----
E=ep(1); A=ep(2); m=ep(3);

b=[ Ex(2)-Ex(1); Ey(2)-Ey(1) ];
L=sqrt(b'*b);
n=b'/L; G=[ n zeros(size(n));
           zeros(size(n)) n ];
Kle=(E*A/L)*[ 1 -1;
              -1 1];
Ke=G'*Kle*G;
Mle=(m*L/6)*[ 2 1;
              1 2];
Me=G'*Mle*G;
%-----end-----
```

I.4.2 Barre 3D :

➤ Syntaxe :

$[K_e, M_e] = \text{bar3e}(ex, ey, ez, ep)$

➤ Description :

bar3e fournit la matrice de rigidité **Ke** et la matrice de masse **Me** pour un élément de barre tridimensionnel. Dont les variables d'entrée

$ex = [x1 \ x2]$

$ey = [y1 \ y2]$

$ez = [z1 \ z2]$

$ep = [E \ A \ \text{Rho} \cdot A]$ fournissent les coordonnées nodales de l'élément $x1, y1, z1, x2$ etc., le module d'élasticité **E**, et l'aire de la section transversale **A** et la masse volumique **Rho**.

➤ Théorie :

La matrice de rigidité globale $[K]^e$ de l'élément est calculée selon la formule suivante :

$$[K]^e = [T]^T [\bar{k}]^e [T] \quad (1.116)$$

Avec :

$$[\bar{k}]^e = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (1.117)$$

De même Nous écrivons la matrice de masse :

$$[M] = [T]^T [\bar{M}] [T] \quad (1.118)$$

La matrice de transformation T s'écrit sous cette forme suivante :

$$T = \begin{bmatrix} C_X & C_Y & C_Z & 0 & 0 & 0 \\ 0 & 0 & 0 & C_X & C_Y & C_Z \end{bmatrix} \quad (1.119)$$

Elle contient les cosinus de direction : $C_X = \cos\theta_x$, $C_Y = \cos\theta_y$, $C_Z = \cos\theta_z$

Où la longueur est égale à :

$$L = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (1.120)$$

➤ Programmation :

```
function [Ke,Me]=bar3e(ex,ey,ez,ep)
% Ke,Me=bar3e(ex,ey,ez,ep)
%-----
% PURPOSE
% calculer la matrice de rigidité et de masse pour element barre a 3
dimension .
%
% INPUT:  ex = [x1 x2]
%         ey = [y1 y2]          element node coordinates
%         ez = [z1 z2]
%
%         ep = [E A m]          E: Module de Young
%                                A: l'aire de la section transversale
%         m=rho*A               Rho : masse volumique
% OUTPUT: Ke : Matrice de rigidité , dim(Ke)= 6 x 6
%-----
E=ep(1); A=ep(2); m=ep(3);
b=[ ex(2)-ex(1); ey(2)-ey(1); ez(2)-ez(1) ];
L=sqrt(b'*b);
n=b'/L; G=[ n zeros(size(n));
            zeros(size(n)) n ];
Kle=E*A/L*[1 -1;-1 1];
Ke=G'*Kle*G;
Mle=(m*L/6)*[2 1;1 2];
Me=G'*Mle*G;
%-----end-----
```

Chapitre II. Méthodes de détection d'endommagement

I.1 Introduction :

De nos jours nous observons un développement et une utilisation croissants de composants et de structures légers, rigides, multicouches et multifonctionnels dans les conceptions techniques modernes. Au cours de la période de fonctionnement, les composants d'une structure peuvent être endommagés suite à plusieurs facteurs externes tels que la surcharge accidentelle, le feu, la corrosion, les charges répétitives, les charges de souffle et les catastrophes naturelles comme les tremblements de terre ; Ces endommagements, s'ils ne sont pas détectés assez tôt, ils s'aggravent généralement avec le temps et finissent par provoquer une défaillance de la structure. Pour éviter ce type de défaillance et afin de garantir l'état de santé structurelle de ces composants et structures il est important d'effectuer une évaluation rapide et correcte des endommagement , cette évaluation doit comprendre l'identification, localisation et la quantification des endommagements si les endommagement sont visible, l' évaluation peut être effectuée assez facilement ,cependant, tout endommagement situé à l'intérieur d'un élément structurel ne peut être évalué que par des essais non destructifs.

Les essais non destructifs peuvent être de deux types : locaux et globaux. Les méthodes locales utilisent la vitesse d'impulsion des ultrasons, les essais thermiques, les essais d'émission acoustique, le champ magnétique, et les courants de Foucault. Ces tests nécessitent généralement une connaissance préalable et précise du voisinage de la section endommagée et souvent, ils doivent être répétés sur l'ensemble de la structure, ce qui peut prendre du temps et être peu coûteux.

En tant que méthode rentable d'évaluation directe ou globale de l'intégrité structurelle, les méthodes de détection des endommagements structurels basées sur les vibrations ont fait l'objet d'une attention croissante pour des applications multiples, nous trouvons que de nombreux chercheurs ont étudié dans l'analyse vibratoire l'effet du changement de fréquence d'une structure à l'état endommagé par rapport à l'état sain.

I.2 Méthodes de détection des endommagements basées sur les vibrations :

Récemment, la possibilité d'utiliser des données vibratoires mesurées pour détecter des changements dans les systèmes structurels en raison de leur endommagement a suscité un intérêt croissant. Plusieurs méthodes d'identification des endommagements par vibration ont été proposées et différentes méthodes peuvent être utilisées pour leur classification le principe sous-jacent de ces méthodes est que la signature vibratoire est un indicateur sensible de l'intégrité physique des structures ;lorsqu'un endommagement surviens dans une structure, les paramètres structurels, tels que la rigidité, la flexibilité et l'énergie de déformation, sont modifiés.et par conséquent les paramètres modaux, tels que la fréquence naturelle, la forme du mode et l'amortissement, seront également modifiés ⁵. Par conséquent, la modification des paramètres structurels et les paramètres modaux peuvent être utilisés comme indicateurs des pour identifier les endommagements dans une structure.

En général, la détection des endommagements peut être définie par quatre niveaux ⁶ comme le montre le schéma suivant :

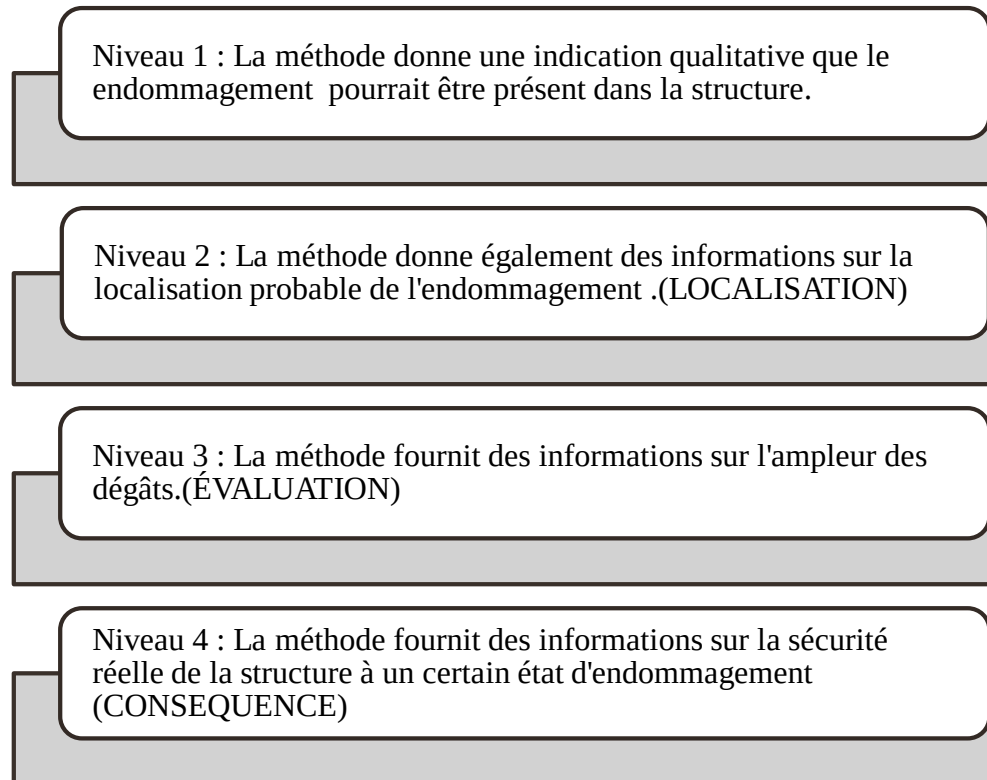


Figure 3. Schéma des 4 niveaux de détection des endommagements

Comme les paramètres modaux sont en relation avec les caractéristiques de rigidité, de masse et d'amortissement alors cette méthode consiste à analyser les changements de ces paramètres suite à un endommagement, on distingue plusieurs méthodes basées sur ces paramètres :

- 1) Méthodes basées sur les changements de fréquence
- 2) Méthodes basées sur les changements de forme de mode
- 3) Méthode de la courbure de la forme du mode
- 4) Méthodes basées sur les changements de la matrice de flexibilité
- 5) Méthodes basées sur l'amortissement
- 6) Méthode basée sur le vecteur résiduel modal

- 7) Méthode de l'énergie de déformation modale
- 8) Méthodes de recalage de la matrice
- 9) Méthodes de réseau neuronal
- 10) Algorithme génétique

I.2.1 Méthodes basées sur les changements de fréquence :

Les fréquences naturelles sont parmi les premières méthodes utilisées dans les applications Structural Health Monitoring (SHM) en raison de leur facilité d'évaluation. Néanmoins, plusieurs études ont confirmé que les méthodes d'identification des endommagements basées uniquement sur les fréquences naturelles peuvent ne pas être appliquées à des structures complexes ou si l'entité endommagée est modeste [Güemes et al.2020].⁷

En général, l'endommagement est un phénomène local, alors que la fréquence naturelle est une caractéristique globale qui n'est pas appropriée pour identifier l'emplacement de l'endommagement, sauf pour des applications simples. Il y a cependant des tentatives pour localiser les endommagements uniquement sur la base de l'information sur les fréquences naturelles. La sensibilité des fréquences naturelles au changement de rigidité est exprimée par [Cawley et Adams 1979]⁸.

De telles méthodes peuvent être trouvées dans les travaux de [Kim et Stubbs 2003]⁹ les deux auteurs ont utilisé les données de fréquence pour détecter des fissures dans les structures de type poutre.

[Yang et le wang 2010]¹⁰ Ont utilisé les vecteurs des fréquences naturelles pour détecter l'endommagement dans un portique avec la nouvelle conception du vecteur de fréquence naturelle (NFV) et du critère d'assurance du vecteur de fréquences naturelles (NFVAC). [Xu, Zhu, Emory 2007]¹¹ on fait une étude expérimentale et numérique de la détection des endommagements structurels à l'aide des changements de fréquences naturelles en modifiant la structure de manière contrôlée et en utilisant les premières fréquences

naturelles mesurées de la structure modifiée. [Xiang et Liang .2012]¹² ont présenté une méthode de détection des endommagements multiples en deux étapes qui estime la quantification du défaut basée sur la fréquence naturelle après la localisation du endommagement par la forme du mode.

I.2.2 Méthodes basées sur les changements de forme de mode :

La méthode la plus simple et la plus simple (après la méthode de la fréquence naturelle) est la méthode dite des déplacements modaux qui utilise les changements de formes de modes qui se produirait à proximité des endommagements, pour ensuite prévoir l'emplacement des endommagements. Sur la base des formes de mode, plusieurs techniques ont été développées au fil des ans pour améliorer les performances de la méthode de déplacement modal.

Un exemple de la méthode est dite -critère d'assurance modale (MAC)- Depuis son introduction [Mahmod 19 97]¹³, qui a utilisé des valeurs MAC et d'autres critères d'assurance similaires pour mesurer la relation entre deux modes . [Fox 1992]¹⁴ a montré que les mesures à un seul chiffre du mode sont comparativement plus efficaces que les autres mesures et insensibles aux défauts d'une poutre ayant une coupe de type scie.

[Kim et al 1992]¹⁵ a utilisé la méthode MAC dans le développement de la méthode Coordinate Modal Assurance (COMAC), qui prend les déplacements des nœuds modaux pour détecter et localiser les endommagements.

Cependant, il n'est pas suffisant de localiser les endommagements en utilisant uniquement ces méthodes. En particulier, le MAC est un peu insensible lorsqu'il y a une répartition expérimentale, les changements de forme de mode sont généralement si petits que la détection des endommagements n'est pas pratique. Les différences graphiques et les changements relatifs dans la forme de mode fournissent le moyen de détecter l'emplacement du l'endommagement lorsque les fréquences de résonance et les formes de mode ont été examinées .

I.2.3 Méthode de la courbure de la forme du mode

Une approche modale populaire qui était introduite par [Pandey et al. 1991]¹⁶ est la méthode des courbures modales qui présente une meilleure stabilité aux endommagements que les valeurs du MAC et du COMAC . De plus, les déformations ont été utilisées pour obtenir les formes de mode de courbure expérimentales au lieu du déplacement ou de l'accélération.

[Qiao ,Lu,Lestari,Wang 2007]¹⁷ ont utilisé cette méthode pour la détection des délaminations dans les plaques composites à l'aide de deux systèmes de mesure un **vibromètre** des capteurs qui mesure les formes des modes de courbure qui sont par la suite mises en œuvre avec quelques algorithmes relativement nouveaux de détection des endommagements pour évaluer l'emplacement et la taille de la délamination.

L'inconvénient de la détection des endommagements basée sur les courbures conduit généralement à une courbe qui présente un pic proéminent à l'emplacement de l'endommagement, mais peut également présenter des valeurs plus petites, ce qui pose des problèmes en applications pratiques [Ciambella et Vestroni ,2015]¹⁸ en utilisant une solution perturbatrice ont réussi à éviter cet inconvénient, ou ils ont prouvé que les courbures modales ne fournissent pas d'informations sur la localisation des endommagements , si elles ne sont pas correctement traitées. Les auteurs ont donc introduit une nouvelle procédure de filtrage pour les courbures , qui conduit à une localisation efficace des endommagements en ne considérant qu'un seul mode.

I.2.4 Méthodes basées sur les changements de la matrice de flexibilité:

Cette méthode a été introduite non seulement pour identifier la présence d'endommagements mais aussi pour les localiser [Pandey et Biswas , 1994]¹⁹ la matrice de flexibilité peut être facilement et précisément estimée à partir de quelques modes de vibration de basse fréquence de la structure, qui peuvent être facilement mesurés. Des modèles analytiques simples de poutres permettent d'étudier l'effet de la présence de endommagements dans une structure sur sa flexibilité. En utilisant ces modèles analytiques, les auteurs ont démontré

l'efficacité de l'utilisation des changements dans la matrice de flexibilité pour détecter et localiser les endommagements.

[Li et al. 2010]²⁰ ont utilisé une nouvelle approche de détection des endommagements structurels sur une poutre simplement soutenue basée sur les changements de la matrice de flexibilité généralisée ;ils ont trouvé que , par rapport à l'approche originale basée sur la matrice de flexibilité, l'effet de la restriction des modes d'ordre supérieur peut être considérablement réduit dans cette nouvelle approche.

Cette méthode été proposé par [Hu · Min Xiao et al 2011]²¹ pour la détection des endommagements des poutres en bois au moyen des courbures de flexibilité modale en utilisant les informations combinées des fréquences et des formes de mode. Ils ont constaté que la flexibilité modale est plus sensible à la description des perturbations locales de la structure car elle combine en même temps les informations des fréquences et des formes de mode.

I.2.5 Méthodes basées sur l'amortissement :

L'existence d'un endommagement dans une structure entraîne généralement des changements dans la capacité d'amortissement de la structure qui dépendent fortement de plusieurs facteurs, tels que la température, l'historique des charges et le traitement pendant la fabrication. Ces facteurs font de l'amortissement un paramètre peu pratique pour détecter les endommagements dans une structure.

[Frizzarin et al. 2008]²¹ ont utilisé pour les structures en béton, une méthode de détection des endommagements dans le domaine temporel, qui est basée sur l'analyse de l'amortissement non linéaire à partir des réponses vibratoires structurelles mesurées. L'efficacité de la méthode proposée a été démontrée à l'aide d'un modèle de pont en béton à grande échelle soumis à différents niveaux des endommagements sismiques causés par des tests sur table vibrante. En appliquant la

technique de signature de décrétement aléatoire, la méthode proposée a réussi à identifier, à partir de ses réponses vibratoires ambiantes, l'amortissement non linéaire du pont associé aux endommagements sismiques. La quantité d'amortissement non linéaire augmente au fur et à mesure que ces endommagements deviennent plus sévères. Et en comparant également les résultats de la détection ces endommagements avec ceux obtenus par des méthodes basées sur la rigidité, ces auteurs ont démontré une forte relation entre l'augmentation de l'amortissement non linéaire et la diminution de la rigidité structurelle associée à l'augmentation des endommagements et leurs gravité .

[Kawiecki,2001]²² a décrit l'application de réseaux d'éléments piézoélectriques liés à la surface pour déterminer les caractéristiques d'amortissement modal d'une structure testée. Ces informations peuvent être utilisées pour la détection des endommagements structurels ou l'évaluation de la dynamique structurelle. Ces caractéristiques d'amortissement, sont obtenues en utilisant des méthodes d'analyse modale standard à partir de la fonction de transfert de fréquence résultante. Comme les endommagements structurels ou matériels sont souvent associés à des changements d'amortissement, la méthode utilisée par cet auteur peut être appliquée à la surveillance de l'état des structures et tester les structures légères et les microstructures.

I.2.6 Méthode basée sur le vecteur résiduel modal

[Yang.Liu 2007]²³ ont étudié les méthodes d'identification des endommagements structurels basées sur le vecteur de force résiduelle. Le vecteur de force résiduelle des nœuds est défini pour localiser les éléments supposés endommagés de manière préliminaire à l'aide du vecteur de force résiduelle. Ensuite, trois techniques de quantification des endommagements sont étudiées pour identifier les endommagements. La première est la solution algébrique de l'équation de la force résiduelle, la deuxième est la technique MREU et la troisième est la méthode de sensibilité à la fréquence naturelle. Une technique d'expansion de forme de mode basée sur le

concept du meilleur vecteur propre réalisable est présentée pour résoudre le problème des mesures incomplètes. Ces méthodes de détection des endommagements sont démontrées sur un exemple numérique et les bruits de mesure.

[Yang .2009]²⁴ a proposé une méthode attrayante du point de vue du calcul pour donner un aperçu de la localisation et de la quantification des endommagements structurels. La méthode proposée utilise la technique de désassemblage matriciel et aborde le problème de la localisation et de niveau des endommagements de manière découplée. Tout d'abord, un schéma est développé pour déterminer la localisation des endommagements en calculant un vecteur de localisation de ces endommagements, qui est dérivé des critères de force résiduelle modale. Une fois l'emplacement déterminé, le niveau d'endommagement correspondant peut-être facilement obtenue uniquement par les processus de division.

1.2.7 Méthode de l'énergie de déformation modale :

Lorsqu'un mode dynamique donné conserve une quantité importante d'énergie de déformation, les données modales de ce mode seront beaucoup plus sensibles aux changements dans la structure. Les changements dans l'énergie de déformation modale peuvent donc être utilisés comme un indicateur utile pour l'identification des endommagements.

[Shi et al. 2000]²⁵ ont présenter une méthode de détection des endommagements structurels basée sur la variation de l'énergie de déformation modale (MSE) avant et après l' endommagement et ils l'ont appliquée sur une structure à portique en acier à une seule baie et deux étages . Les résultats indiquent que l'approche proposée est sensible au bruit, mais qu'elle peut localiser les endommagements simples et multiples.

[Yan et al,2010]²⁶ Ont proposé une méthode de détection des endommagements basée sur un algorithme algébrique efficace de la sensibilité de l'énergie de déformation modale des éléments

afin de détecter l'emplacement et la quantification des endommagements. Les auteurs ont appliqué un certain nombre de scénarios des endommagements supposés pour des poutres simulées numériquement et une structure de cadre bidimensionnelle pour démontrer l'applicabilité de la méthode proposée. Les résultats ont démontré que la méthode proposée est efficace dans une procédure en une étape (non itérative).

I.2.8 Méthodes de recalage de la matrice :

Le recalage du modèle d'éléments finis (FEMU) basée sur la sensibilité est l'une des techniques largement acceptées pour l'identification des endommagements dans les structures. Cette FEMU peut être formulé comme un problème d'optimisation numérique et résolu de manière itérative en faisant un recalage automatique des paramètres inconnus du modèle en minimisant la différence entre les propriétés structurelles mesurées et analytiques.

[Min et al. 2014]²⁷ Ont présenté premièrement une méthode de recalage du modèle FEM basée sur la sensibilité et utilisant les fréquences naturelles et les fréquences nulles. Deuxièmement, une technique de système réduit amélioré et itéré (IIRS) pour réduire le nombre de DOF du modèle MEF. Troisièmement, une matrice d'amortissement est estimée en utilisant les rapports d'amortissement modaux identifiés par une méthode d'ajustement de courbe et les matrices modifiées qui sont obtenues par la mise à jour du modèle et la réduction des DOF. La méthode proposée de mise à jour du modèle FEM est appliquée à une poutre en porte-à-faux réelle fixée à un matériau amortissant sur un côté. Le résultat montre que la méthode proposée peut conduire à une mise à jour précise du modèle des structures amorties.

I.2.9 Méthodes de réseau neuronal :

La détection des endommagements basée sur les réseaux neuronaux consiste généralement en une phase de formation et une phase de reconnaissance. L'algorithme de rétro propagation par

erreur intégrant la méthode du gradient peut être appliqué pour former le réseau neuronal, mais l'efficacité de la formation dépend fortement du taux d'apprentissage. Bien que les techniques d'identification des endommagements par vibration offrent plusieurs avantages, la plupart des algorithmes d'identification des endommagements disponibles échouent lorsqu'ils sont appliqués à des structures pratiques en raison de l'effet des erreurs de mesure, de la nécessité d'utiliser des formes de mode incomplètes, de la restriction des modes et de la nature non unique des solutions.

[Xu, Humar,2006]²⁸ ont présenté un nouvel algorithme robuste en deux étapes qui utilise l'indice de endommagements basé sur l'énergie modale pour localiser les endommagements et une technique de réseau neuronal artificiel pour déterminer le taux des endommagements qui ont été appliqué sur une poutre et dans un modèle similaire d'un pont réel. Les résultats montrent que l'algorithme proposé est assez efficace pour identifier l'emplacement et le taux des endommagements, même en présence d'erreurs de mesure dans les données d'entrée.

[Fang et al ,2005]²⁹ ont mets en œuvre le réseau neuronal pour la détection des endommagements structurels basée sur les FRF(fonctions de réponse en fréquence). Qui ils ont appliqué sur une poutre en porte-à-faux ,les résultats de l'analyse montrent que, dans tous les cas d'endommagements (c'est-à-dire les cas des endommagements formés ,et endommagements non vus, simples et les cas multiples), le réseau neuronal peut évaluer les conditions d'endommagements avec une très bonne précision.

[Rhim et Lee 1995]³⁰ ont fait une étude qui examine la faisabilité de l'utilisation d'un réseau neuronal artificiel en conjonction avec des techniques d'identification de système pour détecter l'existence et identifier les caractéristiques des endommagements dans les structures composites. La méthodologie proposée ici comprend une phase de formation et une phase de reconnaissance. les résultats des tests numériques démontrent la faisabilité de la méthode proposée.

I.2.10 Algorithme génétique :

La détection des endommagements structurels est un problème inverse de l'ingénierie structurelle. Il y a trois questions principales dans cette méthode : l'existence, la localisation et le niveau de endommagements. Les approches habituelles intègrent l'analyse de sensibilité et l'algorithme d'optimisation pour minimiser les écarts entre les données vibratoires mesurées et les données analytiques. Les algorithmes génétiques explorent la région de l'ensemble de l'espace de solution et peuvent obtenir la solution optimale globale.

[Chou,Ghaboussi, 2001]³¹ Ont utilisé les mesures statiques des déplacements à quelques degrés de liberté (DOF) pour identifier les changements des propriétés caractéristiques des éléments structurels tels que le module de Young et l'aire de section transversale, qui sont indiqués par la différence entre les réponses mesurées et calculées. Afin d'éviter les analyses structurelles lors de l'évaluation de l'aptitude, les déplacements au niveau des DOF non mesurés sont également déterminés par l'algorithme génétique . La méthode proposée est capable de détecter l'emplacement approximatif des endommagements, même lorsque des considérations pratiques limitent le nombre de mesures sur site à quelques-unes seulement.

[Hao et al , 2002]³² Ont utilisé un algorithme génétique avec codage des nombres réels qu'ils ont appliqué pour identifier les endommagements structurels en minimisant la fonction objectif, une méthode qui compare directement les changements dans les mesures avant et après l'endommagement. Trois critères différents sont considérés, à savoir les changements de fréquence, les changements de forme de mode et une combinaison des deux. Cet algorithme est appliqué sur une poutre et un cadre testés en laboratoire afin de démontrer la technique proposée. Les résultats numériques montrent que les éléments endommagés peuvent être détectés par l'algorithme génétique, même lorsque le modèle analytique n'est pas précis.

[Perera et Torres ,2006]³³ Ont proposé une approche d'optimisation non classique impliquant

l'utilisation d'algorithmes génétiques (GA) pour localiser les zones endommagées de la structure. La méthode a été vérifiée par un certain nombre de scénarios d'endommagements pour des poutres simulées et par l'utilisation directe de données expérimentales provenant des essais de vibration d'une poutre. Les résultats ont démontré que l'AG proposé permet de localiser les endommagements et de quantifier le taux d'endommagement tout en présentant de nombreux avantages par rapport aux méthodes classiques ; l'influence du bruit dans les données modales a également été prise en compte.

[Tiachacht et al 2018]³⁴ ont présenté une nouvelle méthodologie pour l'identification et la quantification des endommagements dans les structures bidimensionnelles et tridimensionnelles qui est étudiée numériquement en utilisant la méthode des éléments finis (FEM) et le programme Matlab. Ils ont utilisé l'algorithme génétique (AG) et l'indicateur de Cornwell (MCI) comme fonctions objective pour comparer les indicateurs mesurés et calculés. Les résultats de cette analyse montrent que la technique proposée est précise et efficace, par rapport aux autres techniques de la littérature, pour estimer le taux d'endommagement structurels.

L'algorithme génétique (AG) est l'une des méthodes d'optimisation évolutionnaire (OE) les plus utilisées. L'AG est une stratégie qui guide le processus de recherche en imitant la nature. L'objectif est d'explorer efficacement l'espace de recherche pour éviter d'être piégé dans une zone confinée de l'espace de recherche. Il utilise des décisions probabilistes pour la reproduction de la génération suivante et la mutation pour une recherche diversifiée pendant la génération. Ce caractère aléatoire basé sur la probabilité permet d'explorer l'espace de recherche de manière globale il utilise l'expérience de recherche pour guider la recherche. C'est la principale différence avec les optimisations classiques de trajectoires.

Chapitre III. Méthode de Rapport des changements des fréquences locales.

I.1 Introduction

Le changement de comportement global d'une structure doit être relié à des paramètres locaux décrivant l'endommagement. Dans ce but, il est important de sélectionner des indicateurs d'endommagement qui sont sensibles pour caractériser les changements structurels dus aux endommagements. De nombreuses études ont tenté de déterminer et de localiser les endommagements à partir des changements de fréquences propres. Ces méthodes utilisant uniquement les fréquences propres présentent cependant un certain nombre de limites.

Dans ce chapitre, Nous allons utiliser une méthode de détection de défauts dans les structures barres d'où cette méthode utilise le concept du rapport des changements des fréquences locales suite à des défauts de fissurations, de déformations et de ruptures de la structure dont les rigidités de ces deux structures sont affectées. Cette méthode a été bien employée dans le cas d'un seul ou de plusieurs endommagements.

I.2 . Recherche Bibliographique sur la méthode :

Les données modales incluses dans l'analyse d'identification des endommagements ont une grande influence sur la précision des résultats d'identification.

Cependant, la méthode LFCR est une approche qui est basée sur le rapport de changement de fréquence local c'est-à-dire elle utilise ces fréquences pour fournir une précision supérieure afin de localiser le bon emplacement des endommagements dans les

scénarios d'éléments endommagés simples et multiples.

[Khatir et al 2019]³⁵ ont proposé une application basée sur le rapport local de changement de fréquence (LFCR) pour évaluer les endommagements de trois couches (0o/90o/0o) de plaques composites laminées. D'où les auteurs ont utilisé un indicateur pour aider à localiser les éléments endommagés, qu'ils soient uniques ou multiples. Les résultats obtenus indiquent que même lorsqu'il a plusieurs éléments endommagés, l'indicateur LFCR peut détecter ces endommagements avec précision. Et pour plus de précision et prouver que le LFCR est bien meilleur pour l'identification de l'emplacement des endommagements dans les composites stratifiés, un bruit blanc gaussien a été introduit avec différents niveaux. Les résultats obtenus indiquent que même sous un bruit de mesure de 2%, le LFCR peut identifier les endommagements réels avec une grande précision.

[Zenzen et al,2020]³⁵ ont suggéré une approche qui utilise le réseau neuronal artificiel (ANN) combiné avec un nouvel indicateur de l'endommagement modifié, pour identifier les endommagements dans une poutre et une plaque composite à trois couches laminées croisées [0°/90°/0°] présentant des endommagements uniques et multiples. Cette approche basée sur l'analyse modale utilisant la technique de transmissibilité pour améliorer le rapport local de réponse de fréquence (LFCR), L'objectif principal de l'indicateur de l'endommagement proposé est de réduire le nombre de données collectées pour une détection rapide et plus précise, au lieu de collecter toutes les données d'analyse modale, c'est-à-dire les fréquences naturelles, les rapports d'amortissement et les formes de mode, ou d'utiliser l'analyse inverse pour la quantification des endommagements. La fiabilité et la précision de l'application proposée ont localiser et quantifier le taux des endommagements dans les structures composites considérées après l'analyse de quatre scénarios d'endommagements.

[Radzieński et al 2011]³⁶ Ont présenter une nouvelle méthode de détection des endommagements structurels basée sur les paramètres modaux obtenus expérimentalement. La nouvelle méthode est adaptée à la détection des endommagements dus à la fatigue survenant dans une poutre en aluminium. Les endommagements ont été pratiquement réalisés sous forme de coupes de scie de différentes tailles et à différents endroits. La première étape de l'analyse comprenait une tentative d'identification des endommagements à l'aide des indicateurs les plus souvent utilisés, basés sur les paramètres modaux mesurés. À cette fin, une technique spéciale

de traitement du signal a été proposée pour améliorer l'efficacité des indicateurs testés. Cependant, les résultats obtenus n'ont pas été satisfaisants. C'est ce qui a motivé la définition de nouveaux indicateurs d'endommagement (indicateur d'endommagement basé sur le changement de fréquence, méthode hybride de détection d'endommagement), utilisant le changement des fréquences naturelles et toute forme de mode (mesurée ou modélisée), car la mesure des fréquences prend beaucoup moins de temps que la mesure de la forme de mode totale. Il a été démontré que la technique proposée est adaptée à la localisation des endommagements dans les structures de type poutre.

[Dahak et al 2017]³⁷ Ont proposé une autre nouvelle méthode pour localiser la zone des endommagements dans une poutre discrétisée en un certain nombre de zones, où chaque zone a une classification spécifique des quatre premières fréquences naturelles normalisées. La zone endommagée est distinguée uniquement par la classification des fréquences normalisées de la structure. Dans les cas où l'endommagement est symétrique au nœud de vibration, les auteurs ont utilisé la fréquence naturelle normalisée comme deuxième information pour obtenir une localisation plus précise. L'efficacité de cette méthode proposée est démontrée par une simulation numérique avec le logiciel ANSYS et une étude expérimentale avec différents endommagements.

I.3 Présentation de la méthode :

L'équation aux valeurs propres d'une structure à n DDL est écrite comme suit :

$$K\phi = M\phi A \quad (3.1)$$

Quand cette structure est endommagée, les propriétés mécaniques de cette structure changent (résistance et sa rigidité) donc sa matrice de rigidité change. Nous pourrions alors interpréter cette matrice comme la somme des deux matrices de rigidité K_s de la structure saine et $\sum a_i \cdot K_j$ matrice de rigidité élémentaire multipliée par a_i le facteur de diminution de la rigidité en pourcentage c'est-à-dire matrice de rigidité endommagée :

$$K_e = K_s + \sum_{j=1}^L \Delta K_j = K_s + \sum_{j=1}^L a_{ij} \cdot K_j \quad (3.2)$$

En effet, les endommagements ont un impact important sur la rigidité de la structure. Il suffit d'un petit changement dans la rigidité pour provoquer un changement dans les vecteurs propres et les valeurs propres respectivement notés ϕ_i , λ_i .

Si la rigidité de la structure est légèrement perturbée, nous écrivons :

$$[(K + \Delta K) - (\lambda_i + \Delta\lambda_i) \cdot M](\phi_i + \Delta\phi_i) = 0 \quad (3.3)$$

En développant cette formule nous obtiendrons :

$$\begin{aligned} K\phi_i + \Delta K\phi_i + K\Delta\phi_i + \Delta K\Delta\phi_i \\ = \lambda_i M\phi_i + \Delta\lambda_i M\phi_i + \lambda_i M\Delta\phi_i + \Delta\lambda_i M\Delta\phi_i \end{aligned} \quad (3.4)$$

$$\begin{aligned} \phi_i (K + \Delta K) + \Delta\phi_i (K + \Delta K) = \phi_i (\lambda_i M + \Delta\lambda_i M) + \Delta\phi_i (\lambda_i \\ M + \Delta\lambda_i M) \end{aligned} \quad (3.5)$$

$$\Delta\phi_i ((K + \Delta K) - (\lambda_i M + \Delta\lambda_i M)) = \phi_i ((\lambda_i M + \Delta\lambda_i M) - (K + \Delta K)) \quad (3.6)$$

Nous mettons :

$$(K + \Delta K) = K \quad \text{et} \quad (\lambda_i M + \Delta\lambda_i M) = \lambda_i M \quad (3.7)$$

$$\text{Avec : } \frac{1}{\phi_i} = \phi_i^T.$$

$$\Delta \phi_i^* \phi_i^T (K - \lambda_i M) = 0 \quad (3.8)$$

$$\Delta \phi_i (K \phi_i^T - \lambda_i \phi_i^T M) = 0 \quad (3.9)$$

Ou :

$$\Delta \phi_i^T (K \phi_i - \lambda_i \phi_i M) = 0 \quad (3.10)$$

$$\Delta \lambda_i \cdot \phi_i^T \cdot M \cdot \phi_i = \Delta K \cdot \phi_i^T \cdot K \cdot \phi_i \quad (3.11)$$

Donc :

$$\Delta \lambda_i = \frac{\Delta K \cdot \phi_i^T \cdot K \cdot \phi_i}{\phi_i^T \cdot M \cdot \phi_i} = \frac{\sum_{j=1}^L a_j \cdot \phi_i^T \cdot K \cdot \phi_i}{\phi_i^T \cdot M \cdot \phi_i} \quad (3.12)$$

Cette équation donne la variation de la i ème fréquence naturelle.

Nous définissons la notion de la fréquence locale (FL) comme le rapport de l'énergie potentielle de l'élément par rapport à l'énergie cinétique de l'élément pour différents vecteurs propres :

$$LF_{ij} = \frac{\phi_i^T \cdot K \cdot \phi_i}{\phi_i^T \cdot M \cdot \phi_i} \quad (3.13)$$

j : élément et i : mode

Nous pouvons alors calculer les fréquences locales de la structure saine LFu_{ij} et ensuite, celles de la structure endommagée LFd_{ij} .

$$LFu_{ij} = \frac{\phi_i^T \cdot K \cdot \phi_i}{\phi_i^T \cdot M \cdot \phi_i}, LFd_{ij} = \frac{\phi_{id}^T \cdot K \cdot \phi_{id}}{\phi_{id}^T \cdot M \cdot \phi_{id}} \quad (3.14)$$

Afin de localiser les endommagements, Nous utilisons la notion de rapport de changement de fréquence locale (**LFCR**) comme suit :

$$LFCR_{ij} = \frac{|LFd_{ij} - LFu_{ij}|}{LFu_{ij}} \quad (3.15)$$

La méthode **LFCR_{ij}** peut être utilisée pour tous les éléments de la structure et pour un nombre déterminés de modes propres (pour des raisons de précision et d'efficacité on calcule les dix premiers modes).

Si un endommagement se produit dans l'élément **j = p**, la valeur équivalente de **LFCR_{ip}** est la plus grande de toutes celles des autres éléments **j ≠ p**.

I.4 Programmation de la méthode :

Afin de programmer la méthode nous allons d'abord endommager nos éléments selon 3 cas pour chaque cas nous allons augmenter le nombre d'éléments a endommager pour obtenir des fréquences que nous comparons ensuite aux fréquences de notre structure a l'état sain pour ensuite tester la puissance et la précession de cette méthode LFCR dans les trois cas d'endommagements. .

I.4.1 Structure 20 barres :

Chaque élément sélectionné nous l'endommageons d'un certain pourcentage comme le montre le tableau suivant. et a chaque cas nous augmentons le nombre d'éléments endommagés .

20 barres –cas 1	Elément N°	4
	Taux d'endommagement	20%
20 barres-cas2	Elément N°	2
	Taux d'endommagement	15%
	Elément N°	9
	Taux d'endommagement	25%
20 barres –cas 3	Elément N°	8
	Taux d'endommagement	15%
	Elément N°	11
	Taux d'endommagement	20%
	Elément N°	12
	Taux d'endommagement	25%

Tableau 1. Pourcentage des éléments endommagés barre2D

Nous obtiendrons ensuite les fréquences endommagées que nous allons comparer aux fréquences saines pour trouver la différence entre les deux fréquences en pourcentage.

Fréquences naturelles			1	2
FEM			120,096	200,261
Fréquences endommagées	Cas 1		120,096	200,206
		%	0 %	0.027%
	Cas 2		120,001	199,831
		%	0.079%	0.214%
	Cas 3		119,917	200,131
		%	0.149%	0.064%

Tableau 2. Comparaison des fréquences saines et endommagées barre2D

```

[~,Egv]=eigen(K,M); %calculer les vecteurs propres de la structure saine
[~,Egvd]=eigen(Kd,M); %calculer les vecteurs propres de la structure
endommagée
p=1;% mode
pour i=1:p

phi=Egv(:,i);% calculer le i eme vecteur propre sain de chaque elment
phi_d=Egvd(:,i); %calculer le i eme vecteur propre endommagé de chaque
elment
pour j=1:nb_element
% calculer la matrice de rigidite et de masse de la structure saine
[Ke,Me]=bar2e2D(Ex(j,:),Ey(j,:),ep);
ep_e=[E(j) A rho*A];

```

```

%calculer la matrice de rigidite de la structure endommagée
[kd,~]=bar2e2D(Ex(j,:),Ey(j,:),ep_e);

[nie,n]=size(Edof);
t=Edof(:,2:n);

    pour ii = 1:20 (nombre d'elements )
K=zeros(nddl);KD=zeros(nddl);
M=zeros(nddl); % Matrices des zéros
K(t(ii,:),t(ii,:)) = K(t(ii,:),t(ii,:))+Ke;
KD(t(ii,:),t(ii,:)) = KD(t(ii,:),t(ii,:))+kd;
M(t(ii,:),t(ii,:)) = M(t(ii,:),t(ii,:))+Me;
    Fin
[~,nd]=size(K);
fdof=[1:nd]';
pdof=b(:);
fdof(pdof)=[];

LFu(i,j)=(phi'*K(fdof,fdof)*phi)/(phi'*...
    M(fdof,fdof)*phi); % frequencies locales saines

LFd(i,j)=(phi_d'*KD(fdof,fdof)*phi_d)/(phi_d'*...
    M(fdof,fdof)*phi_d); % frequencies locales endommagées

fin
fin
pour i=1:p
    pour j=1:nb_element
%calculer le rapport de changement des frequencies locales
lfc(i,j)=abs(LFd(i,j)-LFu(i,j))/LFu(i,j);
lfcj(j)=sum(lfc(:,j))/max(i);
    fin
fin

```

I.4.2 Structure 28 barres :

Nous ferons de la même manière que la structure précédente comme le montre le tableau ci-dessous :

28 barres-cas1	Elément N °	6
	Taux d'endommagement	20%
28 barres-cas 2	Elément N°	3
	Taux d'endommagement	15%
	Elément N°	14
	Taux d'endommagement	25%
28 barres-cas2	Elément N °	25
	Taux d'endommagement	15%
	Elément N°	18
	Taux d'endommagement	20%
	Elément N°	17
	Taux d'endommagement	25%

Tableau 3. Pourcentage des éléments endommages barre3D

Après avoir endommagé les éléments selon le tableau précédent nous obtiendrons les fréquences présentées dans ce tableau suivant :

Fréquences naturelles			1	2	3	4	5	6
FEM			24,526	51,761	65,266	100,052	114,281	121,937
fréquences endommagées	Cas 1		24,526	51,650	65,250	97,504	113,582	121,839
		%	0%	0.19%	0.21%	2.54%	0.61%	0.08%
	cas 2		24,526	51,757	65,258	99,976	114,195	121,835
		%	0%	0.21%	0.024%	0.075%	0.524%	0.083%
	Cas3		24,524	51,727	64,887	98,435	111,683	120,186
		%	0.008%	0.065%	0.58%	1.61%	2.27%	1.43%

Tableau 4 : Comparaison des fréquences saines et endommagées barre 3D

```

%conditions aux limites
K(:,b)=[];K(b,:)=[];
M(:,b)=[];M(b,:)=[];
Kd(:,b)=[];Kd(b,:)=[];

[~,Egv]=eigen(K,M); %calculer les vecteurs propres de la structure saine
[~,Egvd]=eigen(Kd,M); %calculer les vecteurs propres de la structure
endommagée
%eleminer le premier vecteur propre
Egv(:,1)=[];
Egvd(:,1)=[];
p=1;
pour i=1:p
phi=Egv(:,i); %calculer le i eme vecteur propre sain de chaque elment
phi_d=Egvd(:,i); %calculer le i eme vecteur propre endommagé de chaque
elment

pour j=1:nb
% calculer la matrice de rigidite et de masse de la structure saine

[k1,m1]=bar3e(Ex(j,:),Ey(j,:),Ez(j,:),ep);
epd=[Ed(j) A rho*A];
% calculer la matrice de rigidite de la structure endommagée

[k_d1,~]=bar3e(Ex(j,:),Ey(j,:),Ez(j,:),epd);

[nie,n]=size(Edof);
t=Edof(:,2:n);

pour ii = 1:nie %nie :nombre d'éléments
K=zeros(Ndt);KD=zeros(Ndt);
M=zeros(Ndt); % matrices zeros

K(t(j,:),t(j,:)) = K(t(j,:),t(j,:))+k1;
KD(t(j,:),t(j,:)) = KD(t(j,:),t(j,:))+k_d1;
M(t(j,:),t(j,:)) = M(t(j,:),t(j,:))+m1;

[~,nd]=size(K);
fdof=[1:nd]';
pdof=b(:);
fdof(pdof)=[];

LFu(i,j)=(phi'*K(fdof,fdof)*phi)/(phi'*...
M(fdof,fdof)*phi); % frequencies locales saines

LFd(i,j)=(phi_d'*KD(fdof,fdof)*phi_d)/(phi_d'*...
M(fdof,fdof)*phi_d); % frequencies locales endommagées
fin
fin

pour i=1:p
pour j=1:nb

%calculer le rapport de changement des frequencies locales

```



```

    lfcrl(i,j)=abs(LFd(i,j)-LFu(i,j))/LFu(i,j);
    lfcrl(j)=sum(lfcrl(:,j))/max(i);
  fin
fin

```

I.5 Applications numériques (LFCR)

I.5.1 Treillis 2 D :

Dans cette étude, nous avons modélisé la structure treillis en éléments finis, Chaque nœud de cet élément fini possède deux degrés de liberté : déplacement normal à la barre selon l'axe x et un déplacement autour de l'axe y (voir Figure 4. Treillis 2D 20 éléments Figure 4). Le nombre total de degrés de liberté (DDL) de cette structure est 18.

Soit une structure treillis discrétisée en 20 éléments finis, dont les caractéristiques sont comme suit :

$$L = 2 \text{ m} ; E=6.910^{10} \text{ N/m}^2 ; \text{Rho}=2740 \text{ Kg/m}^3 ; A=1*10^{-4} \text{ m}^2$$

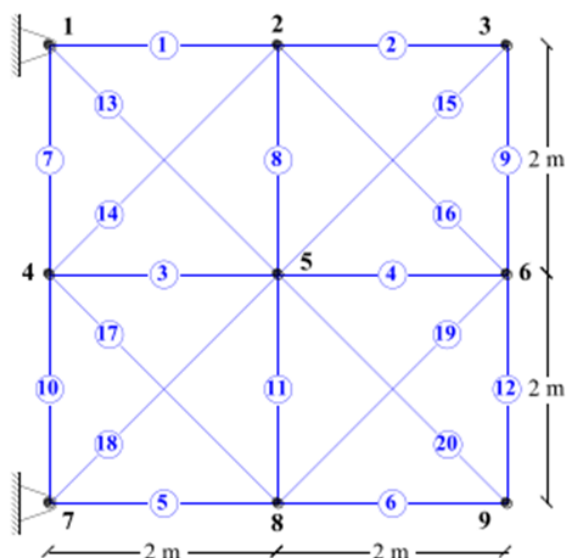


Figure 4. Treillis 2D 20 éléments

Pour le premier cas nous endommageons de 20 % l'élément 4, l'histogramme représentatif des valeurs du LFCR pour chaque élément est donné sur la figure suivante :

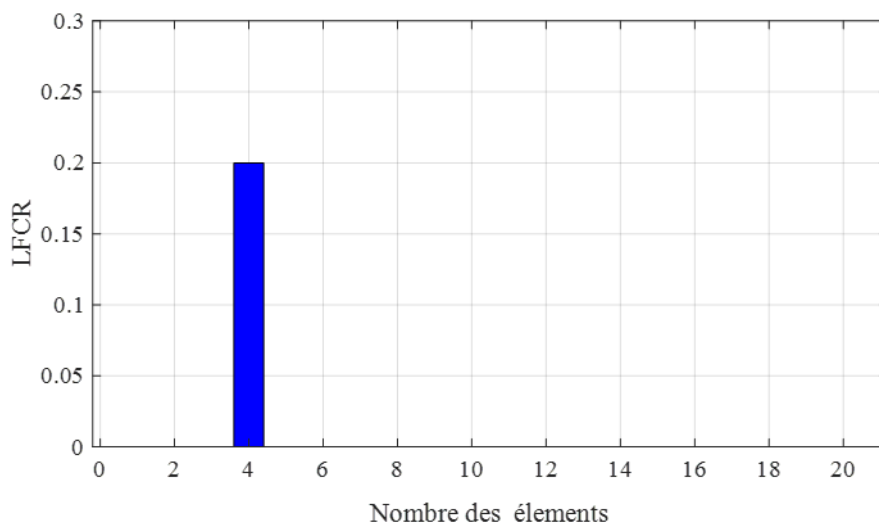


Figure 5: Histogramme des valeurs LFCR d'endommagement de l'élément 4

Pour le 2^{ème} cas nous endommageons de 15 % l'élément 2 et de 25% l'élément 9. Nous obtiendrons par la suite l'histogramme représentatif des valeurs du LFCR comme suit :

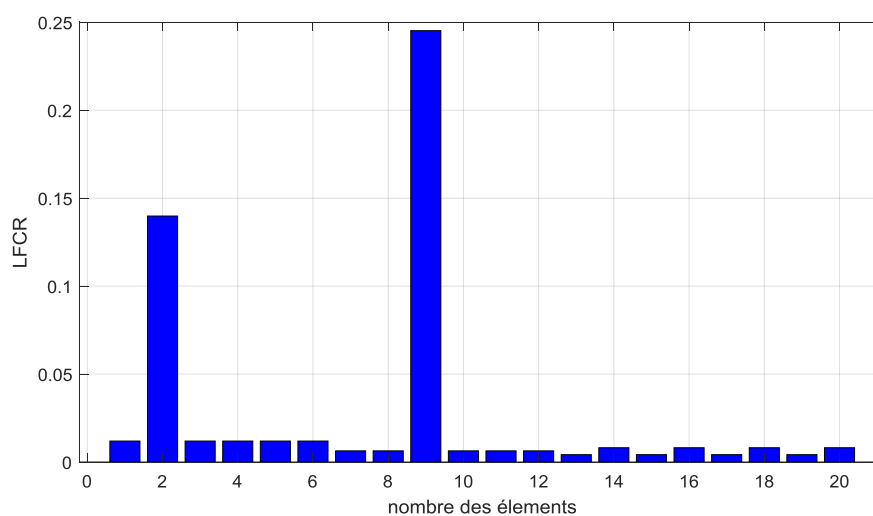


Figure 6: Histogramme des valeurs LFCR pour l'endommagement des éléments 2,9

Pour le 3^{ème} cas nous endommageons de 15 % l'élément 8 et de 20 % l'élément 11 et de 25% l'élément 12. Nous obtiendrons l'histogramme représentatif des valeurs du LFCR comme suit :

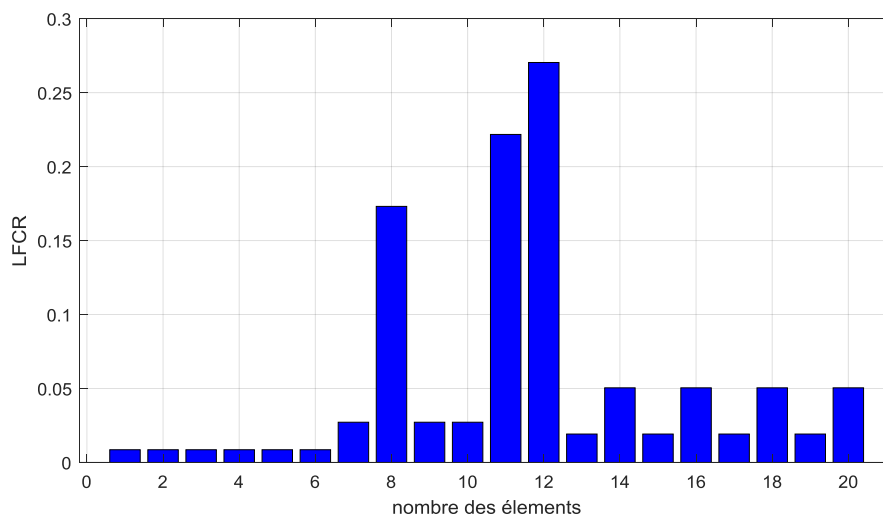


Figure 7: Histogramme des valeurs LFCR pour l'endommagements des éléments 8,11,12

A partir des figures nous remarquons que la valeur du LFCR de l'élément endommagé est la plus grande de toutes, quel que soit sa position sur le treillis et le pourcentage

d'endommagement.

I.5.2 Treillis 3D :

Dans cette étude, nous avons modélisé la structure treillis 3 D en éléments finis, Chaque nœud de cet élément fini possède trois degrés de liberté : déplacement selon l'axe x et déplacement selon l'axe y et selon l'axe z. Le nombre total de degrés de liberté (DDL) de cette structure est 12.

Soit une structure treillis discrétisée en 20 éléments finis, dont les caractéristiques sont comme suivies :

$$A = 0.8529 \times 10^{-2} \text{ m}^2 ; E = 2.1 \times 10^{11} \text{ N/m}^2 ; \text{Rho} = 7830 \text{ Kg/m}^3 ;$$

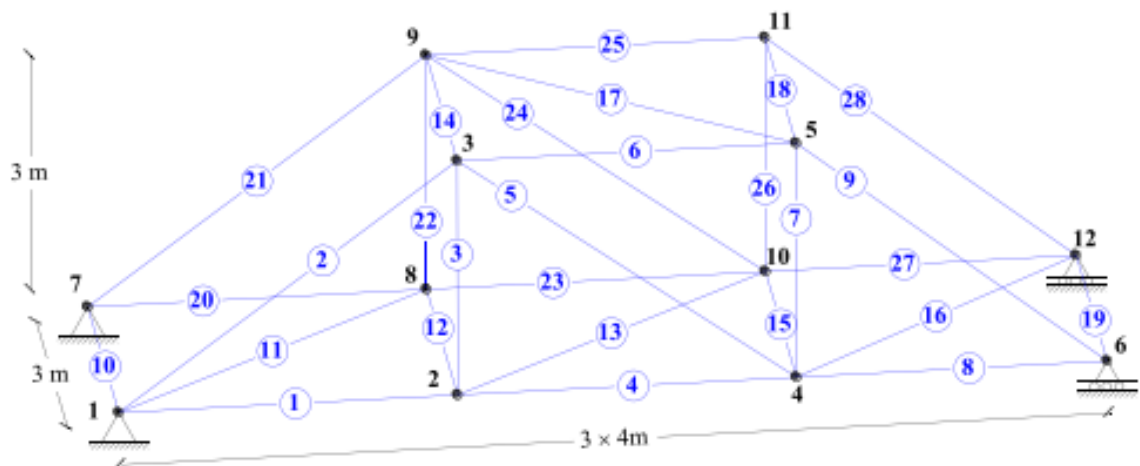


Figure 8: Treillis 3 D 28 éléments

Pour le premier cas nous endommageons de 20 % l'élément 6, l'histogramme représentatif des valeurs du LFCR pour chaque élément est donné sur les FIG

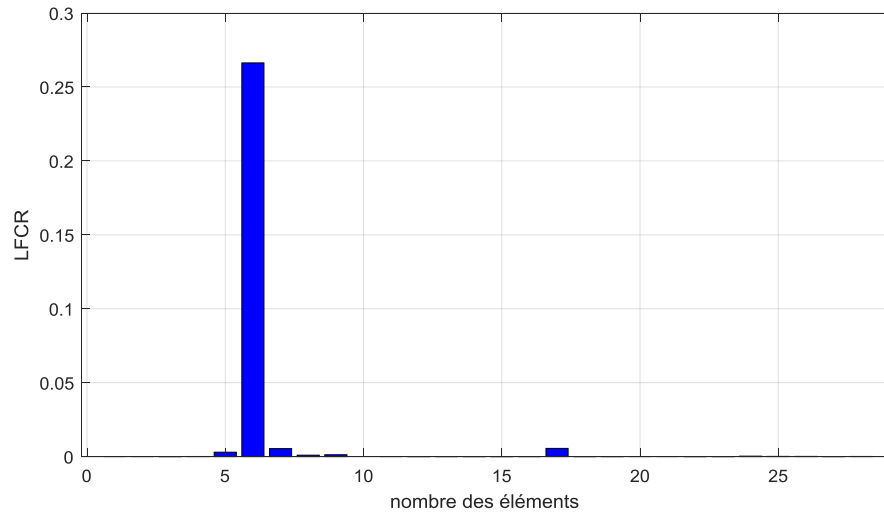


Figure 9: Histogramme des valeurs LFCR pour l'endommagement de l'élément 6

Pour le 2^{ème} cas nous endommageons de 15 % l'élément 3 et de 25% l'élément 14. Nous obtiendrons l'histogramme représentatif des valeurs du LFCR comme suit :

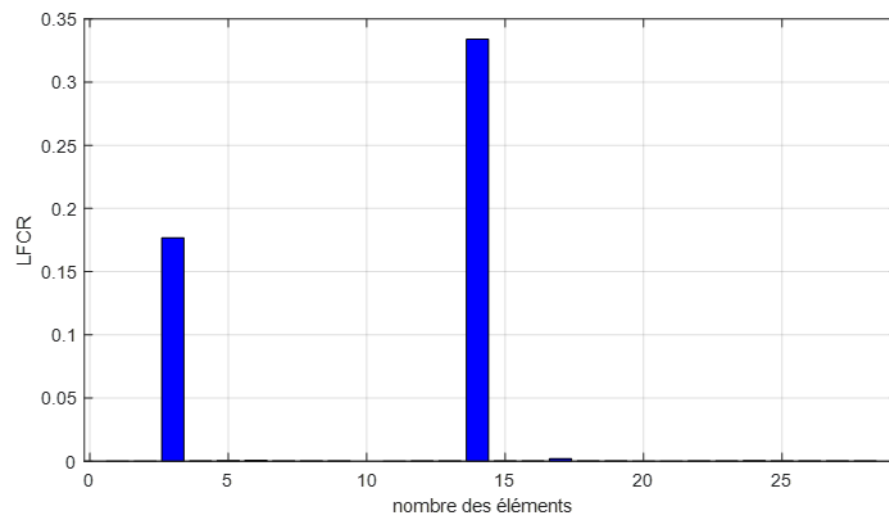


Figure 10: Histogramme des valeurs LFCR pour l'endommagement des éléments 3 et 14

Pour le 3^{ème} cas nous endommageons de 15 % l'élément 25 et de 20 % l'élément 18 et de 25% l'élément 17 ,nous obtiendrons par la suite l'histogramme représentatif des valeurs du LFCR comme suit :

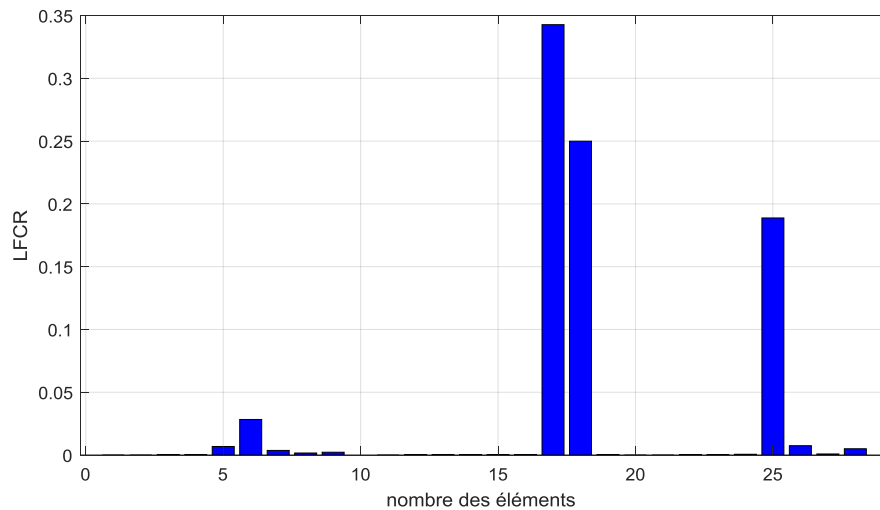


Figure 11: Histogramme des valeurs LFCR pour l'endommagement des éléments 25 et 18 et 17

I.6 Conclusion sur les résultats :

A partir des figures nous remarquons que la valeur du LFCR de l'élément endommagé est la plus grande de toutes, quel que soit sa position sur le treillis et le pourcentage d'endommagement. Les figures 5,6,7,9,10,11 montrent que les valeurs de LFCR des éléments endommagés sont toujours les plus grandes. D'après les résultats obtenus, nous pouvons conclure que la méthode de variation du rapport des changements des fréquences locales LFCR localise bien les endommagements dans les structures treillis.

Chapitre IV. Méthodes d'optimisation et quantification

I.1 Introduction :

La SHM est le meilleur moyen d'identifier et de classer les endommagements. L'intégrité structurelle d'un composite peut être évaluée à l'aide de deux composants principaux, à savoir un ensemble d'accéléromètres et des techniques de calcul. En utilisant ces méthodes, les endommagements peuvent être détectés mais cela demande plus de temps. Le temps nécessaire pour évaluer le niveau d'endommagement peut être optimisé en améliorant la précision et l'efficacité des techniques de calcul. Les techniques de calcul aident à la classification, à la prédiction et à l'optimisation du temps nécessaire à la détection des endommagements parmi ces techniques les plus utilisées pour l'optimisation sont respectivement le réseau neuronal, l'algorithme génétique, l'analyse en composantes principales et l'optimisation.

Les techniques d'estimation des endommagements à partir de données modales sont souvent basées sur des méthodes de quantification du modèle. Ces méthodes reposent sur un modèle paramétrique de la structure et sur la minimisation d'une fonction objective basée sur l'erreur entre les données mesurées et les prédictions du modèle. Le succès de la méthode de quantification du modèle par éléments finis (FE) dépend de la précision du modèle FE, de la qualité des essais modaux, de la définition du problème d'optimisation et de la capacité de l'algorithme d'optimisation.

La nature a évolué sur des centaines de millions d'années, montrant une efficacité et une magie

parfaites. Les gens apprennent beaucoup à partir de l'étude des systèmes naturels et les utilisent pour développer des nouveaux algorithmes et modèles pour résoudre des problèmes complexes.

L'optimisation est une branche des mathématiques, cherchant à analyser et à résoudre analytiquement ou numériquement les problèmes qui consistent à déterminer le meilleur élément d'un ensemble, au sens d'un critère quantitatif donné. Ce mot vient du latin optimum qui signifie le meilleur.

Un algorithme d'optimisation est une procédure mathématique qui permet d'obtenir les minimums (ou maximums) d'une fonction réelle (que l'on appelle fonction objective). De même, le problème inverse de la détection des endommagements (identification de la présence, de l'emplacement et de la taille des endommagements) dans une structure peut être modélisé en utilisant des techniques d'optimisation et d'identification des paramètres. Le problème inverse de la détection des endommagements peut être modélisé par un problème inverse et la présence d'incertitude. Dans le problème direct, étant donné les conditions aux limites et la présence d'endommagements, on obtient la distribution de la quantité d'intérêt dans toute la structure. Dans le problème inverse, on modélise une procédure de localisation des endommagements dans la structure à partir de certaines informations sur la quantité d'intérêt à certains endroits particuliers. De plus, les problèmes directs et inverses sont aléatoire, par conséquent, un certain type de traitement de l'aléatoire doit être effectué au niveau des variables des problèmes. Il est tout aussi important, compte tenu de la modélisation directe des problèmes, d'utiliser des méthodes numériques, telles que la méthode des éléments finis (FEM).

I.2 Généralités sur les algorithmes d'optimisation :

Une conception optimale naïve est obtenue en comparant quelques (jusqu'à une dizaine) de solutions alternatives créées en utilisant une connaissance a priori du problème. La faisabilité de chaque solution de conception est d'abord étudiée. Ensuite, une estimation de l'objectif sous-jacent (coût, profit, etc.) de chaque solution est comparée et la meilleure solution adoptée.

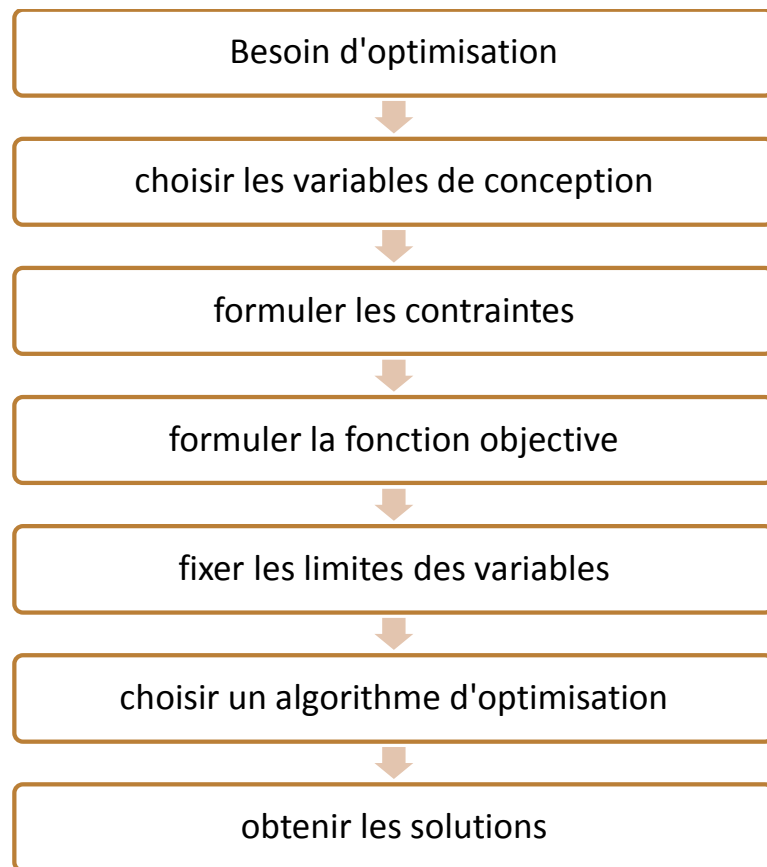


Figure 12. Processus de l'algorithme de l'optimisation

Il est impossible d'appliquer une seule procédure de formulation optimale du problème pour tous les problèmes de conception technique, car l'objectif d'un problème de conception et les paramètres de conception associés varient d'un produit à l'autre, différentes techniques sont utilisées pour problèmes différents. L'objectif de la formulation est de créer un modèle mathématique du problème de conception optimale, qui peut ensuite être résolu à l'aide d'un algorithme d'optimisation. La Figure 12 montre les étapes habituellement impliquées dans une conception optimale formulation.

I.2.1 Variable du problème :

La formulation d'un problème d'optimisation commence par l'identification des variables de conception sous-jacentes, qui varient principalement au cours du processus d'optimisation. Un problème de conception implique généralement de nombreux paramètres de conception, dont certains sont très sensibles au bon fonctionnement de la conception. Ces

paramètres sont appelés variables de conception ou du problème nous pourrions les représenter par une seule variable ou bien plusieurs.

La première règle d'or de la formulation d'un problème d'optimisation est de choisir le moins de variables de conception possible. Le résultat de cette procédure d'optimisation peut indiquer s'il faut inclure davantage variables de conception dans une formulation révisée ou de remplacer certaines variables de conception précédemment considérées par de nouvelles variables de conception.

Nous présentons N variables réelles, rassemblées en un vecteur $[x_1, x_2 \dots \dots x_n]$, ces variables ne seront pas autorisées à prendre n'importe quelle valeur, mais devront satisfaire des contraintes

I.2.2 Contraintes :

Les contraintes représentent certaines relations fonctionnelles entre les variables de conception et d'autres paramètres de conception satisfaisant certains phénomènes physiques et certaines limitations de ressources. La nature et le nombre de contraintes à inclure dans la formulation dépendent de l'utilisateur. Ces contraintes peuvent avoir des expressions mathématiques exactes ou non.

Par exemple, la contrainte maximale est une contrainte d'une structure. Si une structure a une forme régulière, il existe une relation mathématique exacte entre la contrainte maximale et les dimensions. Mais en cas de forme irrégulière un logiciel de simulation par éléments finis peut être nécessaire pour calculer la contrainte maximale. Les deux types de contraintes suivants ressortent de la plupart des considérations :

1. Les contraintes de type inégalité.
2. Les contraintes de type égalité

a) Les contraintes d'inégalité :

Stipulent que les relations fonctionnelles entre les variables sont soit supérieures, soit inférieures, soit égales, une valeur de la ressource.

Exemple :

La contrainte $\sigma(x)$ développée n'importe où dans un composant doit être inférieure ou égale à la résistance admissible du matériau.

$$\sigma(x) \leq S_{Admissible}$$

Certaines contraintes peuvent être de type plus grand que / ou égal à.

b) Les contraintes d'égalité :

Les contraintes d'égalité stipulent que les relations fonctionnelles doivent correspondre exactement à une valeur de ressource.

Exemple :

La déviation $\delta(x)$ d'un point du composant doit être exactement égale à 5 mm. Alors $\delta(x) = 5$. Il est très difficile de traiter les contraintes d'égalité dans les algorithmes. Dans de tels cas, la contrainte d'égalité est relaxée en incluant deux contraintes d'inégalité comme indiqué ci-dessous.

Exemple :

Auparavant,

$$\delta(x) = 5$$

Maintenant, il est changé en contraintes d'inégalité comme indiqué ci-dessous :

$$\delta(x) \geq 4,$$

$$\delta(x) \leq 6$$

I.2.3 Fonctions objectives :

Les algorithmes d'optimisation peuvent être divisés en deux catégories : ceux qui essaient de trouver les meilleures valeurs des fonctions objectives f et ceux qui optimisent des ensembles F de fonctions cibles.

- **Fonctionne unique :**

Dans le cas de l'optimisation d'un critère unique f , un optimum est soit son **maximum**, soit son **minimum**, en fonction de ce que l'on recherche. Si nous possédons une usine de fabrication et que nous devons affecter les commandes entrantes aux machines, nous le ferons de manière à minimiser le temps nécessaire pour les exécuter. D'autre part, nous organiserons l'achat de matières premières, l'emploi du personnel et le placement des publicités de manière à maximiser notre profit. Sur le site optimisation globale, il est de convention que les problèmes d'optimisation soient le plus souvent définis comme des minimisations et si un critère f est soumis à la maximisation, nous minimisons simplement sa négation $(-f)$. La formulation mathématique générale du problème peut être effectuée selon l'équation suivante :

Un optimum local(maximum et minimum) $\hat{x}_i \in X$ d'une fonction (objective).

$f: X \rightarrow \mathbb{R}$ est un élément d'entrée(max) avec $f(\hat{x}_i)_{max} \geq f(x)$ et élément (min)
 $f(\hat{x}_i)_{min} \leq f(x)$ pour tout x voisin de $\hat{x}_i$

Si $X \subseteq \mathbb{R}^n$ on peut écrire :

$$f(\hat{x}_i)_{min} \leq f(x) \leq f(\hat{x}_i)_{max} , \quad \forall i \in [1, n] \quad (4.1)$$

- **Fonctionne multiple :**

Les techniques d'optimisation globale ne sont pas seulement utilisées pour trouver les maximum ou les minimum de fonctions f uniques. Dans de nombreux problèmes de conception

ou de prise de décision du monde réel, elles sont plutôt appliquées à des ensembles F constitués de $n = |F|$ fonctions objectives f_i .

Chacune représentant un critère à optimiser. L'optimisation multi-objectifs consiste souvent à faire des compromis entre des objectifs contradictoires. Si nous revenons à notre exemple d'usine, nous pouvons spécifier les objectifs suivants qui sont tous soumis à l'optimisation :

- Minimiser le temps entre une commande entrante et l'expédition du produit correspondant..
- Maximiser le profit.
- Minimiser les coûts de publicité, de personnel, de matières premières, etc.
- Maximiser la qualité du produit.
- Minimiser l'impact négatif sur l'environnement.

Les deux derniers objectifs semblent clairement en contradiction avec la minimisation des coûts. Entre les coûts personnels, le temps nécessaire à la production et la qualité du produit, il devrait y avoir une sorte de relation (contradictoire). Les influences mutuelles exactes entre les objectifs peuvent apparemment devenir compliquées et ne sont pas toujours évidentes.

La méthode la plus simple pour définir ce qui est optimal consiste à calculer une somme pondérée $g(x)$ de toutes les fonctions $f_i(x) \in F$.

Chaque objectif f_i est multiplié par un poids w_i représentant son importance. L'utilisation de poids signés permet aussi de minimiser un objectif et d'en maximiser un autre. Nous pouvons, par exemple, appliquer un poids $w_a = 1$ à une fonction objectif f_a et le poids $w_b = -1$ au critère f_b . En minimisant $g(x)$, nous minimisons en fait la première fonction objective et maximisons la seconde. Si nous maximisons $g(x)$, l'effet serait inversé et f_b serait minimisé et f_a

serait maximisé. Dans tous les cas, les problèmes multi-objectifs sont réduits à des problèmes mono-objectifs par cette méthode.

$$g(x) = \sum_{i=1}^n w_i f_i(x) = \sum_{\forall f_i \in F} w_i f_i(x) \quad , \quad \forall i \in [0, 1] \quad (4.2)$$

I.2.4 Limites des variables :

La dernière tâche de la procédure de formulation consiste à fixer les limites minimales et maximales de chaque variable de conception. Certains algorithmes d'optimisation n'ont pas besoin de cette information. Dans ces problèmes, les contraintes entourent complètement la région réalisable. D'autres problèmes exigent que l'algorithme de recherche se situe à l'intérieur de ces limites.

En général, toutes les N variables de conception doivent se trouver à l'intérieur des limites minimales et maximales, comme suit.

$$x_i^L \leq x_i \leq x_i^U \quad \text{pour } i = 1, 2, 3, \dots, N$$

Dans tout problème donné, la détermination des variables est limitée et peut s'avérer difficile. Une façon de remédier à cette situation est de deviner la solution optimale et de fixer les limites minimales et maximales de manière à ce que la solution optimale se situe entre ces deux limites.

Si l'une des variables de conception correspondant à la solution optimale se trouve sur ou près de la limite minimale ou maximale, la limite choisie peut être simulé à nouveau à l'algorithme d'optimisation.

Une fois les quatre tâches susmentionnées accomplies, le problème d'optimisation peut être écrit mathématiquement dans un format spécial, connu sous le nom de format de programmation

non linéaire (NLP). C'est à dire résumer toute la procédure sous forme des vecteurs pour les variables et fonctions pour la contrainte.

I.2.5 Les types des algorithmes d'optimisation :

En général, les algorithmes d'optimisation peuvent être divisés en deux classes de base : les algorithmes **déterministes** et les algorithmes **probabilistes**.

Les algorithmes déterministes : sont le plus souvent utilisés si une relation claire existe entre les caractéristiques des solutions possibles et leur utilité pour un problème donné. Dans ce cas, l'espace de recherche peut être exploré efficacement en utilisant, par exemple, une méthode de type schéma "diviser pour régner ". Si la relation entre une solution candidate et son "aptitude"(fitness) ne sont pas si évidentes ou trop compliquées, ou si la dimensionnalité de l'espace de recherche est très élevée, il devient plus difficile de résoudre un problème de manière déterministe. Tenter de le faire pourrait aboutir à l'énumération exhaustive de l'espace de recherche, ce qui n'est pas faisable même pour des problèmes relativement petits.

C'est alors que les algorithmes probabilistes entrent en jeu. Les premiers travaux dans ce domaine, qui est devenu l'un des domaines de recherche les plus importants en optimisation, ont débuté il y a environ 55 ans.

Une famille particulièrement pertinente d'algorithmes probabilistes sont les approches basées sur la méthode de Monte Carlo. Elles échangent la garantie de la correction de la solution pour un temps d'exécution plus court. Cela ne signifie pas que les résultats obtenus à l'aide de ces approches sont incorrects. Ils peuvent simplement ne pas correspondre aux optimaux globaux. D'un autre côté, une solution un peu inférieure au meilleur possible est meilleure que celle qui nécessite 10100 ans pour être trouvée.

Les heuristiques utilisées dans l'optimisation globale sont des fonctions qui aident à décider laquelle d'un ensemble de solutions possibles doit être examinée ensuite. D'une part, les algorithmes déterministes utilisent généralement des heuristiques afin de définir l'ordre de traitement des solutions candidates. Les méthodes probabilistes, quant à elles, peuvent ne prendre en compte dans les calculs ultérieurs que les éléments de l'espace de recherche qui ont été sélectionnés par l'heuristique.

I.2.6 Espace de recherche :

L'espace de recherche G d'un problème d'optimisation est l'ensemble de tous les éléments g qui peuvent être traités par les opérations de recherche, le type de solutions candidates dépend du problème à résoudre.

Comme il existe de nombreuses applications différentes de l'optimisation, il existe de nombreuses formes d'espaces de problèmes. Il serait compliqué de développer des opérations de recherche pour chaque nouvel espace de problème rencontré. Une telle approche serait non seulement une source d'erreurs, mais elle rendrait également très difficile la formulation de lois générales et la consolidation des résultats. Au lieu de cela, nous réutilisons souvent des espaces de recherche bien connus pour de nombreux problèmes différents. Dans ce cas, il suffit de définir une correspondance entre l'espace de recherche et l'espace du problème. Bien que cela ne soit pas toujours possible, cela nous permet d'utiliser davantage de logiciels prêts à l'emploi dans de nombreux cas.

I.2.7 Codage binaire :

Les fonctions de test des algorithmes génétiques ont généralement été conçues avec des propriétés dans l'espace numérique qui rendent difficile la localisation de la solution optimale à l'aide des techniques d'optimisation traditionnelles. Dans certains cas, l'utilisation du **codage Gray**³⁸ a permis d'améliorer les performances de la recherche génétique. Cependant, le codage Gray produit une cartographie de fonction différente qui peut avoir moins de valeurs optimales locales et des relations d'hyperplans relatives différentes.

Valeurs décimale	Code binaire	Code Gray
0	000	000
1	001	001
2	010	011
3	011	010
4	100	111
5	101	110
6	110	101
7	111	100

Table 1. Conversion du code binaire vers le code Gray

Certaines fonctions de test des algorithmes génétiques sont considérablement modifiées par

le codage Gray, de sorte que les méthodes d'optimisation locale sont souvent plus performantes que les algorithmes génétiques. Il est important de mentionner que le codage des variables n'est pas absolument nécessaire. Il existe certaines études où les AG sont directement utilisés sur les variables elles-mêmes.

Les chaînes de caractères codées binaires comportant des 1 et des 0 sont les plus utilisées. La longueur de la chaîne est généralement déterminée en fonction de la précision souhaitée de la solution.

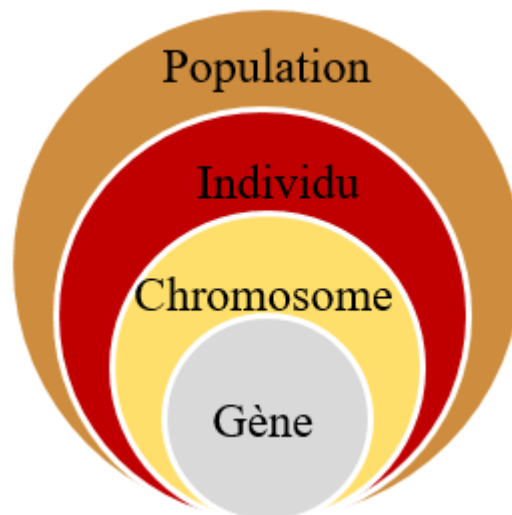


Figure 13: Les cinq niveaux d'organisation de l'algorithme d'optimisation

- **Gène** : chaque paramètre de la solution est assimilé à un gène. Les valeurs que ce gène prend sont des allèles
- **Chromosome** : on peut regrouper les paramètres similaires dans le même chromosome.
- **Individu** : chaque individu est représenté par un ensemble de chromosomes.
- **Population** : Une population est un ensemble d'individus ou d'éléments qui possèdent une ou plusieurs caractéristiques en commun, qui les regroupent.

I.3 Les paramètres d'un algorithme d'optimisation :**Approximation Initiale**

Pour initialiser l'algorithme, il est nécessaire d'avoir une approximation initiale à la solution x_0 . (Point de départ). Le choix d'une bonne approximation initiale conditionne la convergence ou pas à la solution.

La taille de la population :

- Si elle est grande :

-La diversité augmente.

-La convergence vers un optimum local diminue.

- Si elle est petite :

-La probabilité de s'attarder sur les minima locaux est grande.

-Selon le cas elle est entre 25 et 100

Nombre d'Itérations :

Un algorithme d'optimisation utilise un processus récursif, calcule une nouvelle approximation (itération) à la solution réelle jusqu'à ce que les critères de convergence soient atteints. En programmation, c'est une boucle de répétition où la nouvelle approximation est construite à partir des approximations antérieures.

Vitesse de convergence :

Quand on parle de convergence proche d'une solution, on parle de la vitesse à laquelle les termes de l'itération approchent sa limite.

En général, les ordres de convergences sont linéaires , quadratiques , cubiques , quartiques ... Une méthode d'optimisation avec un ordre de convergence supérieur arrive à la solution avec peu d'itérations. Le choix d'une méthode avec une haute convergence est important pour les problèmes d'une certaine taille ou avec de multiples paramètres. Si un algorithme ne converge pas, ça ne veut pas dire qu'il n'existe pas de solution. Il n'existe aucun algorithme universel dont la convergence soit garantie, en général il dépend du choix de l'initialisation x_0 et des propriétés de la fonction.

Le taux de croisement :

- L'opérateur de croisement est appliqué avec une probabilité P_c .
- Plus le taux est élevé plus de nouveaux sont introduits.

Le taux de Mutation

- L'opérateur de mutation est appliqué avec une probabilité P_m .
- Si ce taux est grand, la recherche devient purement aléatoire.
- S'il est faible la population est moins diversifiée et en plus il y a risque de stagnation.

Critère d'arrêt :

Pour arrêter le processus de calcul. Il existe plusieurs critères d'arrêt. Les plus utilisées :

- Un taux minimum qu'on désire atteindre d'adaptation de la population au problème N_{max}
- Un certain temps de calcul ne pas dépasser
- $||x_{n+1} - x_n|| < \varepsilon$ Différence entre deux approximations successives.
- $\varepsilon \in \mathbb{R}$ c'est une tolérance et elle est choisie en fonction du type de problème. En général, c'est une valeur négligeable ($\varepsilon \approx 10^{-4} \cdot 10^{-6}$)

les algorithmes d'optimisations sont considérés comme un processus de recherche utilisé en (Aéronautique, Génie civil, Génie Mécanique et Energétique, Génie Chimique, Génie Informatique et Télécommunication, Génie Electrique, etc. ...)pour trouver une solution exacte ou approximative à des problèmes d'optimisation et de recherche. Les algorithmes d'optimisation sont les principaux responsables de l'efficacité de la détection des endommagements par vibration. Et ils ont l'avantage d'être simples à mettre en œuvre. La modélisation est un outil puissant qui permet en particulier d'optimiser un dispositif avant de le fabriquer. Les algorithmes d'optimisation permettent de s'affranchir en grande partie de la méthode d'essai et erreur.

I.4 Algorithme d'essaim de tuniciers (TSA) :

Les problèmes d'optimisation objective, tels que la minimisation de la consommation de temps, de l'énergie, des coûts et des erreurs ou la maximisation de l'efficacité, des performances et de la qualité d'un processus, sont couramment rencontrés dans les applications du monde réel. Récemment, plusieurs chercheurs ont adopté une nouvelle famille d'algorithmes d'optimisation appelés algorithmes méta heuristiques (MA). Ces algorithmes effectuent une recherche aléatoire dans l'espace des caractéristiques pour obtenir une solution optimale parmi diverses solutions, qui sont principalement inspirées par la nature. Parmi un grand nombre de MA, certains sont populaires, tels que l'optimisation de la flamme de la mite (MFO) , l'algorithme d'optimisation de la baleine (WOA) , l'optimisation du sinus cosinus (SCA) , l'algorithme d'optimisation de la mouette (SOA) , l'algorithme du troupeau de krill et l'optimiseur d'accouplement des bernacles (BMO) , Algorithme d'essaim de tuniciers (TSA) et de pissenlit (DO), car ils sont simples, efficaces et robustes pour trouver des solutions optimales

L'algorithme de l'essaim de tuniciers (TSA) est une nouvelle méthode de recherche puissante inspirée par le comportement étrange des tuniciers (un animal invertébré marin) tel que la montre la Figure 14 pendant la recherche de nourriture. Les tuniciers sont bioluminescents et produisent une lumière bleu-vert pâle qui peut être vue à plus de plusieurs mètres. Les tuniciers sont formés

cylindrique, ouverte à une extrémité et fermée à l'autre. Chaque tunicier a une taille de quelques millimètres. Il existe une tunique gélatineuse commune à tous les tuniciers, qui permet d'unir tous les individus. Cependant, chaque tunicier aspire individuellement l'eau de la mer qui l'entoure et produit une propulsion par jet par son extrémité ouverte par des siphons auriculaires. Cette propulsion est puissante pour migrer les tuniciers verticalement dans l'océan, et ils adoptent deux stratégies principales lors de la recherche de nourriture : la **propulsion par jet** et l'**intelligence en essaim**. La plupart des algorithmes d'optimisation aboutissent à de nouvelles solutions basées sur la solution précédente. Ces stratégies sont appliquées de manière aléatoire aux solutions actuelles afin d'obtenir la meilleure solution.

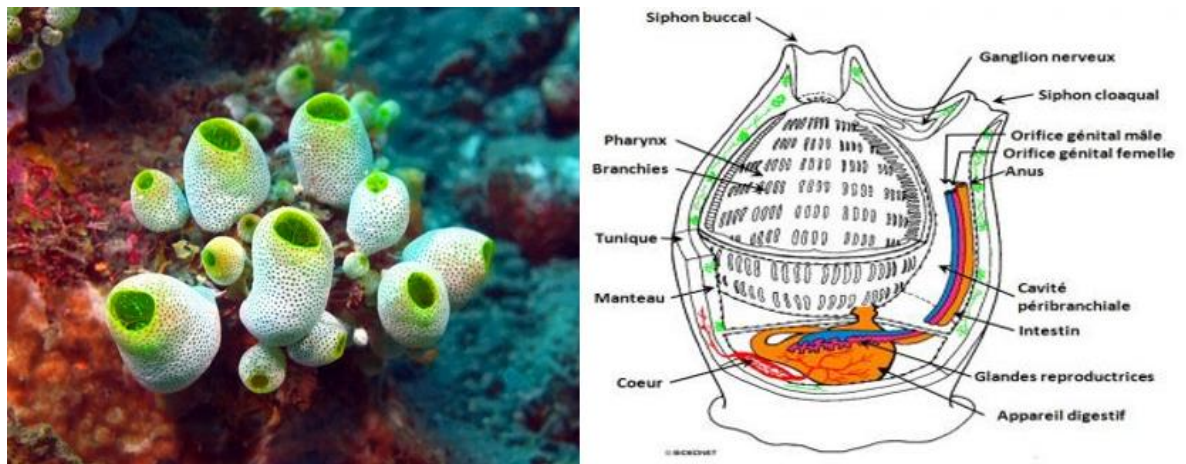


Figure 14. L'image réel des Tuniciers et leur schéma

[Houssein et al , 2021]³⁹ a mené une étude qui intègre un algorithme d'essaim de tuniciers (TSA) avec un opérateur d'échappement local (LEO) pour surmonter les faiblesses du TSA original. La stratégie LEO dans TSA-LEO empêche la déflation (diminution) de la recherche dans TSA et améliore le taux de convergence et l'efficacité de la recherche locale des agents de l'essaim. L'efficacité de la TSA-LEO proposée a été vérifiée sur la suite de tests CEC'2017, et ses performances ont été comparées à celles de sept algorithmes méta heuristiques (MA). Les comparaisons ont révélé que LEO aide significativement TSA en améliorant la qualité de ses solutions et en accélérant le taux de convergence. TSA-LEO a ensuite été testé sur un problème du monde réel, Un ensemble de mesures d'évaluation bien connues a été utilisé pour valider la

performance et la qualité de la segmentation du TSA-LEO proposé. Le TSA-LEO proposé surpasse les autres algorithmes MA en termes de fitness, de rapport signal/bruit maximal, de similarité structurelle des caractéristiques et de résultats de segmentation.

[Kaur et al ,2020]⁴⁰ Ont proposé un algorithme qui imite la propulsion par jet et les comportements d'essaim des tuniciers pendant la navigation et la recherche de nourriture. D'où ils ont évalué la performance de TSA sur soixante-quatorze problèmes de test de référence en utilisant l'analyse de la sensibilité, de la convergence et de l'évolutivité ainsi que le test ANOVA. Ensuite ils ont comparé l'efficacité de cet algorithme à celle de plusieurs approches méta heuristiques reconnues, sur la base des solutions optimales générées. Et d'autre part ils ont exécuté l'algorithme proposé sur six problèmes de conception technique avec et sans contraintes afin de vérifier sa performance. Les résultats de la simulation démontrent que TSA génère de meilleures solutions optimales en comparaison avec d'autres algorithmes compétitifs et qu'il est capable de résoudre des études de cas réels ayant des espaces de recherche inconnus.

[Amirbahador Arabali et al 2022]⁴¹ ont réalisé une modification sur l'algorithme TSA en introduisant un algorithme d'optimisation adaptative de l'essaim de tuniciers (ATSA) qui comporte deux phases principales à chaque itération : **la recherche** autour de l'espace de recherche basée sur un tuniciers sélectionné de manière aléatoire et **l'amélioration** de la recherche en utilisant la position du meilleur tuniciers. Cette étude a été appliquée sur une fondation étalée avec une analyse de sensibilité qui a été menée pour étudier l'effet des paramètres du sol sur le coût total et les émissions de CO2 intégrées de la fondation. Les résultats obtenues sont comparés à ceux de TSA ainsi qu'à certains algorithmes d'optimisation efficaces. Il a été démontré que, par rapport aux autres algorithmes concurrents, l'ATSA est supérieur et peut produire de meilleures solutions optimales.

Un algorithme d'essaim de tuniciers amélioré (ITSA) a été utilisé par [Li et al, 2021]⁴² pour résoudre et optimiser le problème de répartition dynamique des émissions économiques (DEED). L'objectif de l'optimisation DEED est de réduire le coût du combustible et les émissions polluantes

du système électrique. Trois systèmes d'essai contenant 5, 10 et 15 unités de générateur sont utilisés pour vérifier les performances de résolution de l'ASTI. Les résultats des tests montrent que cet algorithme peut fournir un plan d'ordonnancement compétitif pour les systèmes de test contenant différentes unités et il donne le schéma de répartition dynamique optimal du point de vue économique et environnemental pour obtenir une stratégie de répartition plus précise.

[Khajehzadeh et al 2022]⁴³ proposent un algorithme méta heuristique efficace pour résoudre des problèmes d'optimisation globale et optimiser deux structures d'ingénierie géotechnique couramment rencontrées : les murs de soutènement en béton armé et les fondations à faible profondeur. Deux nouvelles équations pour les procédures de mise à jour de la position du leader et des suiveurs ont été introduites dans l'optimisation adaptative par essais salpêtriens (ASSA). Ce changement a amélioré les capacités d'exploration de l'algorithme tout en l'empêchant de converger prématurément. Des fonctions de test de référence ont été utilisées pour confirmer les performances de l'algorithme proposé, et les résultats ont été comparés à l'ASSA et à d'autres algorithmes d'optimisation efficaces. Dans les procédures d'analyse et de conception, les états limites géotechniques et structurels ont été utilisés. Selon les résultats de la simulation, ASSA surpasse les modèles alternatifs et démontre sa capacité à produire de meilleures solutions optimales.

I.4.1 Présentation de l'algorithme TSA :

- **Prévention des conflits (mélange) entre les agents :**

Pour éviter les conflits entre agents lors de la recherche de meilleures positions, les nouvelles positions des agents sont calculées comme suit :

$$\vec{A} = \frac{\vec{G}}{\vec{M}} \quad (4.3)$$

$$\vec{G} = c_2 + c_3 - \vec{F} \quad (4.4)$$

$$\vec{F} = c_1 \cdot \vec{F} \quad (4.5)$$

Où :

$\vec{A}$: est un vecteur des nouvelles positions des agents.

$\vec{G}$: est la force de gravité.

$\vec{F}$: représente le flux d'eau dans l'océan profond,

c_1 , c_2 **et** c_3 : sont trois nombres aléatoires situés dans l'intervalle [0, 1].

Les forces sociales entre les agents sont stockées dans un nouveau vecteur $\vec{M}$, représenté comme suit :

$$\vec{M} = [P_{min} + c_1 \cdot P_{max} - P_{min}] \quad (4.6)$$

Où P_{min} et P_{max} représentent les vitesses initiale et subordonnée pour réaliser une interaction sociale.

- **Mouvement vers la direction du meilleur voisin :**

Après avoir évité le conflit entre voisins, les agents de recherche se déplacent vers la direction du meilleur voisin :

$$\vec{PD} = |X_{best} - r_{rand} \cdot \vec{P_p}(x)| \quad (4.7)$$

Où :

$\overrightarrow{PD}$: est la distance entre la source de nourriture et l'agent de recherche,

x : indique l'itération courante.

$X_{best} = \overrightarrow{FS}$: est la position de la source de nourriture, c'est-à-dire, optimum source de nourriture.

Le vecteur $\overrightarrow{P_p(x)}$: indique la position du tunicier .

r_{rand} : est un nombre aléatoire dans l'intervalle [0, 1].

- **Converger vers le meilleur agent de recherche :**

L'agent de recherche peut maintenir sa position vers le meilleur agent de recherche (c'est-à-dire la source de nourriture).

$$\overrightarrow{P_p(x)} = \begin{cases} \overrightarrow{FS} + \vec{A} \cdot \overrightarrow{PD} & , si r_{rand} \geq 0.5 \\ \overrightarrow{FS} - \vec{A} \cdot \overrightarrow{PD} & , si r_{rand} < 0.5 \end{cases} \quad (4.8)$$

Ou :

$\overrightarrow{P_p(x)}$: est la position actualisée du tunicier par rapport à la position de la source de nourriture $\overrightarrow{FS}$.

- **Comportement de l'essaim**

Afin de simuler mathématiquement le comportement de l'essaim de tunicier, les deux premières meilleures solutions optimales sont sauvegardées et mettent à jour les positions d'autres agents de recherche en fonction de la position des meilleurs agents de recherche. La formule suivante est proposée pour définir le comportement en essaim de l'ascidie plissée :

$$\overrightarrow{P_p(x+1)} = \frac{\overrightarrow{P_p(x)} + \overrightarrow{P_p(x+1)}}{2 + c_1} \quad (4.9)$$

Il y a quelques points importants à propos de l'algorithme TSA qui sont décrits comme suit :

- $\vec{A}$, $\vec{G}$, et $\vec{F}$ aident les solutions à se comporter de manière aléatoire dans un espace de recherche donné et responsables afin d'éviter les conflits entre les différents agents de recherche.
- La possibilité d'améliorer les phases d'exploration et d'exploitation est par les variations des vecteurs $\vec{A}$, $\vec{G}$, et $\vec{F}$.
- Les comportements de propulsion et d'essaimage de l'ascidie dans un espace de recherche donné définissent le comportement collectif de l'algorithme TSA.

I.4.2 Les étapes et l'organigramme de la TSA :

Pour clarifier la TSA, les principales étapes et l'organigramme données ci-dessous illustrent en détail son déroulement original.

Étape 1 : Initialiser la première population de tuniciers P^p

Étape 2 : Définir les paramètres initiaux et le nombre le plus élevé d'itérations (maximal).

Étape 3 : Calculer la valeur de fitness de chaque agent d'exploration(recherche).

Étape 4 : Après avoir calculé la valeur de fitness, Exploré le meilleur agent de recherche dans l'espace de recherche fournie.

Étape 5 : Mise à jour des position de chaque agent de recherche en utilisant l'équation (4.8).

Étape 6 : Ajuster les nouveaux agents de recherche qui ont été mis à jour dans un espace de recherche donné

Étape 7 : Calculer la valeur d'aptitude (fitness) de l'agent de recherche mise à jour. S'il existe une meilleure solution que la solution optimale précédente, alors mettre à jour Pp et enregistrer cette meilleure solution dans Xbest

Étape 8 : Si le critère de fin est satisfait, alors les processus s'arrêtent. Sinon, itérer les étapes 5 à 8.

Étape 9 : Déclarer la meilleure solution optimale (Xbest), qui est obtenue jusqu'à présent.

Programme sur Matlab :

```
% Déclarer les agents de recherche, le maximum d'itérations ,la limite
supérieure et la limite inférieure de l'espace de recherche
%fonction objective (fitness)
function
[Score,Position,Convergence,BEST]=TSA(Search_Agents,Max_iterations,Low
erbound,Upperbound,dimensions,objective)
tic;

Position=zeros(1,dimensions);
Score=inf;
% initialiser la premeiere population des tunicier et definir les
parametres initiaux
Positions=init_TSA(Search_Agents,dimensions,Upperbound,Lowerbound);
Convergence=zeros(1,Max_iterations);
Uj
t=0;
%itérations
while t<Max_iterations
    for i=1:size(Positions,1)

%les limites de l'espace de recherche
        Flag4Upperbound=Positions(i,:)>Upperbound;
        Flag4Lowerbound=Positions(i,:)<Lowerbound;

Positions(i,:)=(Positions(i,:).*(~(Flag4Upperbound+Flag4Lowerbound)))+
Upperbound.*Flag4Upperbound+Lowerbound.*Flag4Lowerbound;

%fonctionne objective
```

```

        fitness=objective(Positions(i,:));

        if fitness<Score
            Score=fitness;
            Position=Positions(i,:);
        end

    end

%les vitesses initiales et subordonnée pour réaliser une interaction sociale.

        xmin=1;
        xmax=4;
        %la force entre les agents
        xr=xmin+rand()*(xmax-xmin);
        xr=fix(xr);

        for i=1:size(Positions,1)
            for j=1:size(Positions,2)

%vecteur de la nouvelle position des agents
                A1=((rand()+rand())-(2*rand()))/xr;

                c2=rand();
                if(i==1)
                    c3=rand();
                    if(c3>=0)
%distance entre la nourriture et l'agent de recherche
                        d_pos=abs(Position(j)-c2*Positions(i,j));
%position du tunicier
                        Positions(i,j)=Position(j)+A1*d_pos;
                    else
%distance entre la nourriture et l'agent de recherche
                        d_pos=abs(Position(j)-c2*Positions(i,j));
%position du tunicier
                        Positions(i,j)=Position(j)-A1*d_pos;

            end
        end

```

```

else

    c3=rand();
    if(c3>=0)
        d_pos=abs(Position(j)-c2*Positions(i,j));
        Pos(i,j)=Position(j)+A1*d_pos;
    else
        Pos(i,j)=Position(j)-A1*d_pos;
    end

end

% Comportement de l'essaim

    Positions(i,j)=(Pos(i,j)+Positions(i-1,j))/2;

end

end

end

t=t+1;
% la convergence (fitness) des meilleurs agents de recherche
Convergence(t)=Score;
[t Score];
% la meilleure position des meilleurs tunicier
BEST(t,:)=Position;
end
end

```

I.5 Optimiseur de pissenlit (DO) :

DO est un nouvel algorithme d'optimisation bio inspiré par l'intelligence en essaim pour résoudre les problèmes d'optimisation continue appelé l'optimiseur de pissenlit (Dandelion Optimizer) . DO simule le processus de vol longue distance des graines de pissenlit tel que le montre la Figure 15 en s'appuyant sur le vent, qui est divisé en trois étapes. Au cours de la phase ascendante, les graines s'élèvent en spirale sous l'effet des tourbillons venant du haut ou dérivent localement en communautés selon les différentes conditions météorologiques. Dans la phase de descente, les graines volantes descendent régulièrement en ajustant constamment leur direction dans l'espace global. Au stade de l'atterrissage, les graines se posent à des endroits choisis au hasard pour se développer.



Figure 15: L'image réel du Pissenlit et son schéma

Cet nouvel algorithme d'intelligence artificielle Inspiré par le comportement du semis du pissenlit, a été proposé en 2017 par [Li et al]⁴⁴ pour l'optimisation globale de fonctions complexes. Dans cet algorithme, la population de pissenlits est divisée en deux sous-populations, et les différentes sous-populations auront des comportements de semis différents. De plus, une autre méthode de semis est conçue pour sortir de l'optimum local. Afin de démontrer la validation de l'AD (Dandelion Algorithm). L'algorithme proposé a été comparé avec d'autres algorithmes existants, notamment l'algorithme de la chauve-souris, l'optimisation

par essaims de particules et l'algorithme des feux d'artifice améliorés. Les simulations montrent que l'algorithme proposé semble bien supérieur aux autres algorithmes.

Dans le même temps, l'algorithme proposé peut être appliqué à l'optimisation du machine learning extrême (ELM) pour les problèmes de classification biomédicale, et l'effet est considérable.

[Zhu et al .2019]⁴⁵ ont proposé un algorithme de pissenlit avec mutation basée sur la probabilité (DAPM). D'où les mutations gaussiennes et de Lévy sont utilisées de manière interchangeable en fonction d'un modèle de probabilité donné. Les auteurs ont examiné trois modèles de probabilité, à savoir les modèles linéaire, binomial et exponentiel. Les expériences montrent que le DAPM obtiens de meilleures performances globales sur les fonctions de test standard que le DA.

En 2020 , pour améliorer la capacité d'exploration de l'algorithme de pissenlit et réduire la probabilité qu'il tombe dans un optimum local,[Han et al]⁴⁶ ont proposer d'ajouter à l'AD un nouveau mécanisme de compétition avec retour d'informations historiques. Plus précisément, la valeur d'aptitude de chaque pissenlit de la génération suivante, qui est calculée par prédiction linéaire, est comparée au meilleur pissenlit actuel, et le perdant est remplacé par un nouveau rejeton. Ce travail améliore son processus de génération de progéniture en exploitant les informations historiques avec un algorithme d'estimation de la distribution. Trois modèles d'information historique sont conçus. Il s'agit du meilleur, du pire et d'un modèle hybride de retour d'informations historiques. Les résultats expérimentaux montrent que les algorithmes proposés surpassent DA et ses variantes, et ils sont supérieurs ou compétitifs par rapport à neuf algorithmes participants évalués sur 28 fonctions de CEC2013. Enfin, les algorithmes proposés et les résultats démontrent leur efficacité sur quatre problèmes du monde réel et indiquent qu'ils ont de meilleures performances que leurs homologues.

[Zhao et al. 2021]⁴⁷ ont décrit la trajectoire d'une graine dans les phases de descente et d'atterrissage par un mouvement brownien (mouvement désordonné et aléatoire) et une marche aléatoire de Levy (succession de pas très courts dans toutes les directions) en utilisant des fonctions de référence de la CEC2017 sur 4 problèmes d'optimisation du monde réel pour évaluer les performances de DO, y compris la précision de l'optimisation, la stabilité, la convergence et l'évolutivité, en les comparant à 9 algorithmes méta heuristiques bien connus inspirés par la nature. Les résultats expérimentaux indiquent que la méthode DO proposée est un optimiseur plus performant avec une optimisation itérative exceptionnelle et une forte robustesse par rapport aux algorithmes bien établis.

Le DO est un algorithme d'optimisation intelligent récemment proposé et montre une excellente performance dans la résolution de problèmes d'optimisation de fonctions. Cependant, comme d'autres algorithmes intelligents, il converge lentement et tombe facilement dans des optima locaux.

I.5.1 Présentation de l'algorithme DO :

Nous présentons brièvement certains opérateurs de DO dont le nombre de graines de pissenlit est calculé comme suit :

$$N_i = \begin{cases} S_{max} + \frac{f_{max} - f(x_i) + \varepsilon}{f_{max} - f_{min} + \varepsilon} & , si N_i > S_{min} \\ S_{min} & , si N_i \leq S_{min} \end{cases} \quad (4.10)$$

où :

S_{max} et S_{min} : sont conçus pour contrôler les nombres maximum et minimum de graines produites

f_{max} : est la valeur d'aptitude maximale de tous les pissenlits,

f_{min} : est la valeur d'aptitude minimale est l'epsilon de la machine pour éviter une erreur de division par zéro,

$f(x_i)$: est la valeur d'aptitude du i-ème pissenlit.

Sur la base de la valeur d'aptitude, les pissenlits sont divisés en deux types en deux types : le meilleur pissenlit et les autres pissenlits. Le site méthode de calcul du rayon d'ensemencement des deux types de pissenlits est différente. Pour le meilleur pissenlit, il est calculé comme suit :

$$R_{BD}(t) = \begin{cases} U_B - L_B & , \quad t = 1 \\ R_{BD}(t-1) * r & , \quad g = 1 \\ R_{BD}(t-1) * e & , \quad g \neq 1 \end{cases} \quad (4.11)$$

où :

U_B, L_B : sont respectivement la limite supérieure et la limite inférieure de l'espace de recherche.

t : signifie la t-ième génération,

r : est un facteur de flétrissement,

e : est un facteur de croissance,

g : reflète la tendance de croissance

Et on calcule la tendance de croissance comme suit :

$$g = \frac{f_{BD}(t) + \varepsilon}{f_{BD}(t-1) + \varepsilon} \quad (4.12)$$

Pour les autres pissenlits, le rayon de semis est calculé de la façon suivante :

$$R_i(t) = \begin{cases} U_B - L_B & , \quad t = 1 \\ w_d \times R_i(t-1) + (||x_{BD}||_\infty - ||x_i||_\infty) & , \quad \text{sinon} \end{cases} \quad (4.13)$$

où w_d est un facteur de pondération calculé comme suit :

$$w_d = 1 - \frac{T_c}{T_{max}} \quad (4.14)$$

T_c et T_{max} sont respectivement le nombre actuel et le nombre maximal d'évaluations de la fonction et le nombre maximum d'évaluations de la fonction, respectivement.

Le meilleur pissenlit a une autre façon de semer, à savoir le semis par mutation. L'opération de mutation spécifique est conçue comme suivante :

$$X'_{BD} = X_{BD} \times 1 + Levy() \quad (4.15)$$

D'où :

$Levy()$ est un nombre aléatoire généré par la distribution de Levy, et il peut être calculé avec le paramètre $\beta = 1,5$.

Dans la DA, il faut que le meilleur emplacement actuel soit toujours conservé pour l'itération suivante. Afin de garder la diversité, les emplacements restants sont sélectionnés en fonction de sur la base de l'opérateur de sélection perturbateur. Pour la localisation X_i la probabilité de sélection p_i est calculée comme suit :

$$p_i = \frac{f_i}{\sum_{n=1}^{SN} f_n} \quad (4.16)$$

$$f_i = |f_i - f_{avg}|,$$

f_i : est la valeur de fitness de la fonction objectif, f_{avg} est la moyenne de toutes les valeurs de fitness de la population dans la génération t .

SN : est l'ensemble de tous les pissenlits (pissenlits, graines normales, et graines de mutation).

Les probabilités de sélection déterminées par cette méthode peuvent donner aux individus, bons ou mauvais, plus de chances d'être sélectionnés pour l'itération suivante, tandis que les individus dont les valeurs de fitness se situent à mi-chemin seront éliminés. Cette méthode permet non seulement de conserver la diversité de la population, mais aussi de refléter une meilleure capacité de recherche globale.

1.5.2 Les Etapes de l'algorithme DO :

Dans cette section, nous présentons DA. Tout comme les autres algorithmes évolutionnaires, DA peut être divisé en quatre parties principales

Etape 1 : INITIALISATION

DA génère aléatoirement N pissenlits dans le domaine de recherche comme population de pissenlits de première génération.

Etape 2 : SEMIS NORMAL

Chaque pissenlit produit des graines de pissenlit dans un certain rayon de semis. rayon de semis. Pour un problème de minimisation, plus la valeur de fitness, plus le nombre de graines

produites est élevé. Le nombre de graines est calculé en fonction de la valeur d'aptitude et le rayon de semis est ajusté dynamiquement. En outre, les méthodes de calcul du rayon d'ensemencement du pissenlit ayant la valeur d'aptitude minimale et des autres pissenlits sont différentes.

Etape 3 : SEMIS PAR MUTATION

La mutation de prélèvement est utilisée pour sortir d'un optimum local, et cette opération de mutation ne concerne que le pissenlit ayant le fitness minimum, qui est appelé le meilleur pissenlit.

Etape 4 : STRATÉGIE DE SÉLECTION

Le meilleur pissenlit est toujours conservé dans la génération suivante. Les N autres pissenlits sont choisis parmi les autres sur la base d'un opérateur de sélection perturbateur.

Programme Matlab :

```
% Déclarer les population ,le maximum d'itérations ,la limite
supérieure et la limite inférieure de l'espace de recherche
%fonction objective (fitness)

[Best_fitness,Best_position,Convergence_curve,BEST]=DO(Popsize,Maxiteration,
LB,UB,Dim,Fobj)
tic;
%initialisation de la population des pissenlits
dandelions=initialization_DO(Popsize,Dim,UB,LB);
%déclarer la fonction objective au fonction des populations
dandelionsFitness = zeros(1,Popsize);
%Déclarer le vecteur convergence en fonction des itérations
Convergence_curve=zeros(1,Maxiteration);
for i=1:Popsize
    % déclarer la fonction objective
    dandelionsFitness(1,i)=Fobj(dandelions(i,:));
end
% Calculer la valeur de fitness pour les pissenlits initiales .
[~,sorted_indexes]=sort(dandelionsFitness);
Best_position=dandelions(sorted_indexes(1),:);
Best_fitness = dandelionsFitness(sorted_indexes(1));
```

```

Convergence_curve(1)=Best_fitness;
t=2;

while t<Maxiteration+1

    %% Rising stage
    beta=randn(Popsize,Dim);
    rand_=randperm(Popsize);
    alpha=rand()*((1/Maxiteration^2)*t^2-2/Maxiteration*t+1); % eq.(8)
in this paper
    a=-1/(Maxiteration^2-2*Maxiteration+1);
    b=-2*a;
    c=1-a-b;
    k=1-rand()*(c+a*t^2+b*t); % eq.(11) in this paper
    if randn()<1.5
        for i=1:Popsize
            lamb=abs(randn(1,Dim));
            theta=(2*rand()-1)*pi;
            row=1/exp(theta);
            vx=row*cos(theta);
            vy=row*sin(theta);
            NEW=rand(1,Dim).*(UB-LB)+LB;

dandelions_1(i,:)=dandelions(i,:)+alpha.*vx.*vy.*lognpdf(lamb,0,1).*(N
EW(1,:)-dandelions(i,:)); % eq.(5) in this paper
            end
        else
            for i=1:Popsize
                dandelions_1(i,:)=dandelions(i,:).*k; % eq.(10) in this
paper
            end
        end
    end
    dandelions=dandelions_1;
    % Check boundries
    dandelions = max(dandelions,LB);
    dandelions = min(dandelions,UB);

    %% Decline stage
    dandelions_mean=sum(dandelions,1)/Popsize; % eq.(14) in this paper
    for i=1:Popsize
        for j=1:Dim
            dandelions_2(i,j)=dandelions(i,j)-
beta(i,j)*alpha*(dandelions_mean(1,j)-
beta(i,j)*alpha*dandelions(i,j)); % eq.(13) in this paper
        end
    end
    dandelions=dandelions_2;
    % Check boundries
    dandelions = max(dandelions,LB);

```

```

dandelions = min(dandelions,UB);

%% Landing stage
Step_length=levy(Popsize,Dim,1.5);
Elite= repmat(Best_position,Popsize,1);
for i=1:Popsize
    for j=1:Dim
        dandelions_3(i,j)=Elite(i,j)+Step_length(i,j)*alpha*(Elite(i,j)-
dandelions(i,j)*(2*t/Maxiteration)); % eq.(15) in this paper
    end
end
dandelions=dandelions_3;
% Check boundries
dandelions = max(dandelions,LB);
dandelions = min(dandelions,UB);

%%
% Calculated all dandelion seeds' fitness values
for i=1:Popsize
    dandelionsFitness(1,i)=Fobj(dandelions(i,:));
end

% Arrange dandelion seeds from good to bad according to fitness
values
 [~,sorted_indexes]=sort(dandelionsFitness);
dandelions=dandelions(sorted_indexes(1:Popsize),:);
SortfitbestN = dandelionsFitness(sorted_indexes(1:Popsize));

%Update the optimal dandelion seed
if SortfitbestN(1)<Best_fitness
    Best_position=dandelions(1,:);
    Best_fitness=SortfitbestN(1);
end

Convergence_curve(t)=Best_fitness;
t = t + 1;
BEST(t,:)=Best_position;
end
time = toc;
end

%
function [z] = levy(n,m,beta)
% beta is set to 1.5 in this paper
num = gamma(1+beta)*sin(pi*beta/2);
den = gamma((1+beta)/2)*beta*2^((beta-1)/2);
sigma_u = (num/den)^(1/beta);

```

```

u = random('Normal',0,sigma_u,n,m);
v = random('Normal',0,1,n,m);
z =u./ (abs(v) .^(1/beta));
end

```

I.6 Tests et résultats :

Dans ce mémoire on a appliqué ces deux méthodes sur les structures qu'on a obtenu dans les chapitres précédents afin de quantifier les taux d'endommagements des éléments endommagés de chaque structure.

I.6.1 Barre 2D 20 éléments :

On présente les résultats des temps qui a été pris pour finaliser les deux algorithmes TSA et DO sur la structure treillis 2D 20 barres dans les 3 cas d'endommagements.

Cases		Cas 1				
P		30			50	100 200
Itérations		50	100	200	50	
DO	13,561495	37,108233	74,4545358	31,6329464	61,2862717	123,3980757
TSA	18,151111	44,7138835	81,2492383	36,872073	68,0030246	130,4990626
Cases		Cas 2				
P		30			50	100 200
Itérations		50	100	200	50	
DO	11,132141	24,2536406	60,1677143	23,1332981	44,0536173	88,6006727
TSA	12,629339	26,8176637	64,4111988	30,5270441	48,9925443	93,5861314
Cases		Cas 3				
P		30			50	100 200
Itérations		50	100	200	50	
DO	13,668658	27,0473962	54,0519124	22,5500973	46,2028558	93,9852907
TSA	18,446007	31,2830088	58,5742109	27,3298961	51,4625203	98,8768497

Tableau 5. Les temps d'exécution des algorithmes TSA et DO sur la barre 2D

D'où :

P : représente l'ensemble des populations, qu'on varie entre [50 ,100 ,200] dans notre test.

Itérations : représente le nombre d'itération appliquées sur la structures avec les différentes valeurs qu'on varie de la même façon que la population [50.10.200].

Il est bien remarqué que TSA prend plus de temps pour l'exécution que l'algorithme DO pour les 3 cas d'endommagements. Le temps d'exécution des deux algorithmes est plus court quand la population est constante($P=30$) et quand les itérations sont variées entre [50.100.200] que quand on varie les populations sur le même intervalle et on donne une valeur constante pour les itérations ($I=50$) le temps d'exécution est long.

C'est bien de varier les itérations et prendre une seule population (constante) pour avoir un temps court d'exécution.

Cependant, on obtient aussi les graphes des localisations et quantification des éléments endommages ainsi la convergence des optimiseurs.

I.6.1.1 Cas 1 un seul endommagement :

Pour un nombre de population constant ($P=30$) et un nombre d'itérations qui vari [50,100,200]

Et pour un nombre d'itérations constant ($I=30$) et un nombre de populations qui vari [50,100,200]

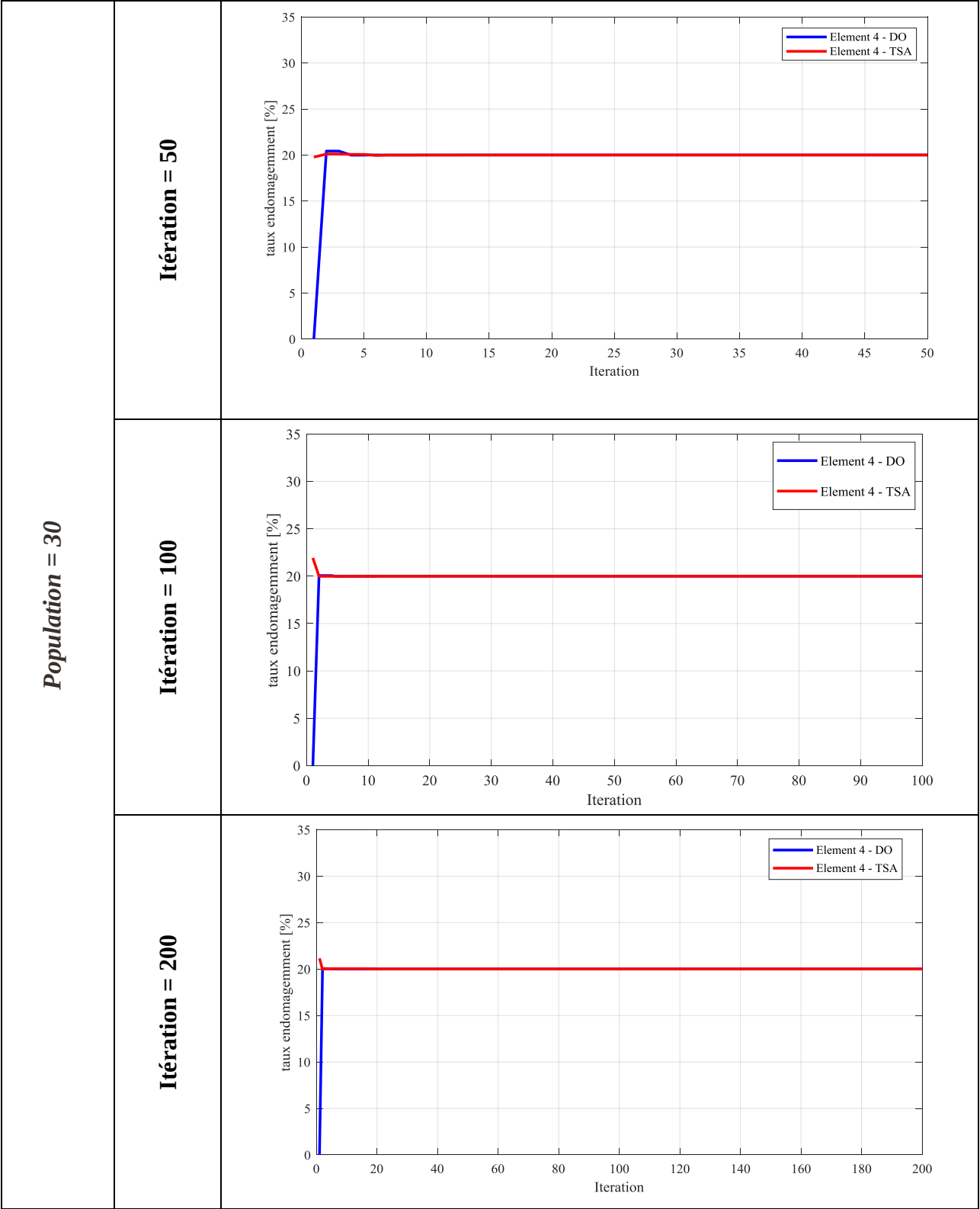


Figure 16. Localisation et quantification des éléments endommagés – cas 1 – barre 2D – population 30.

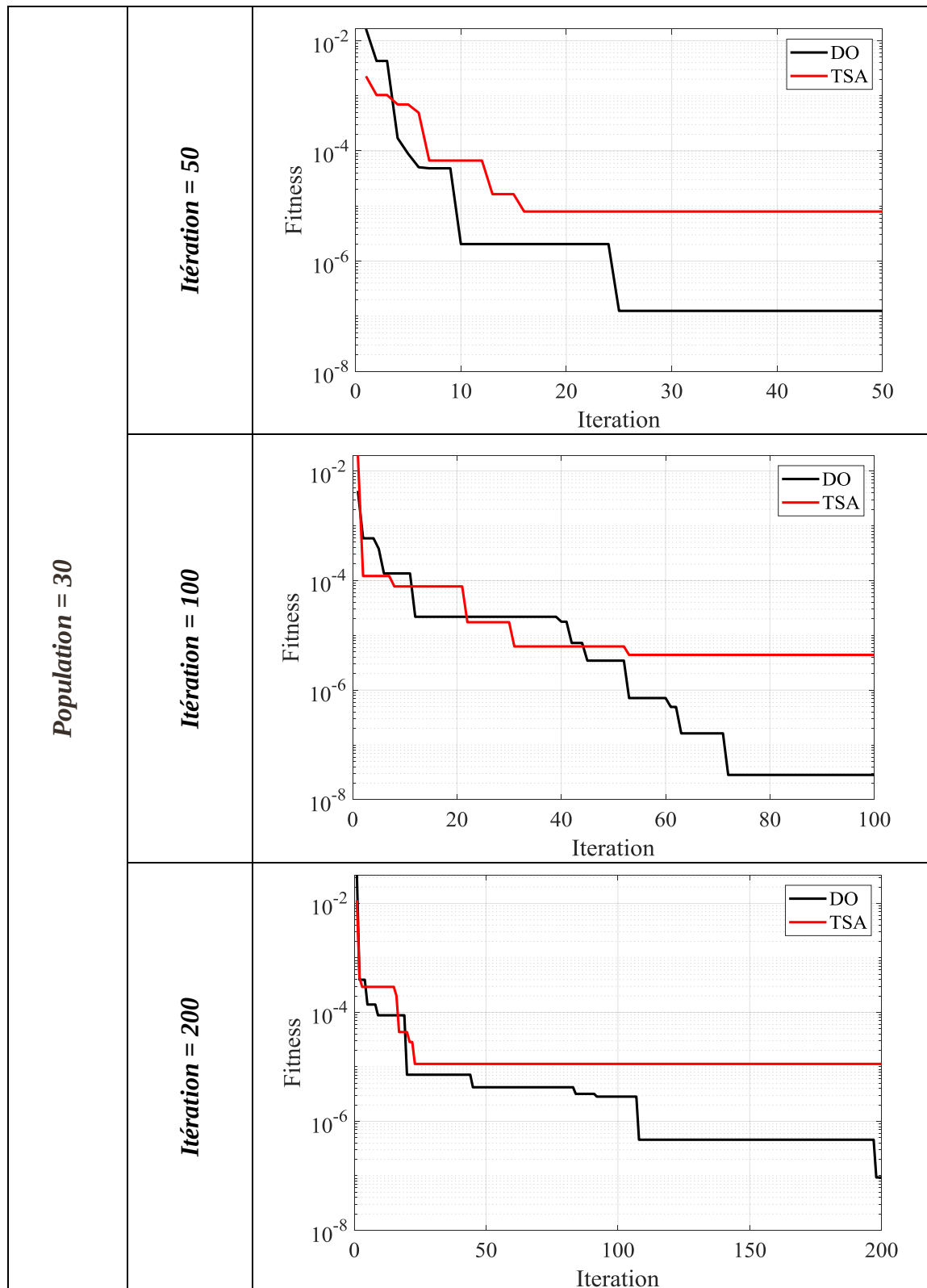
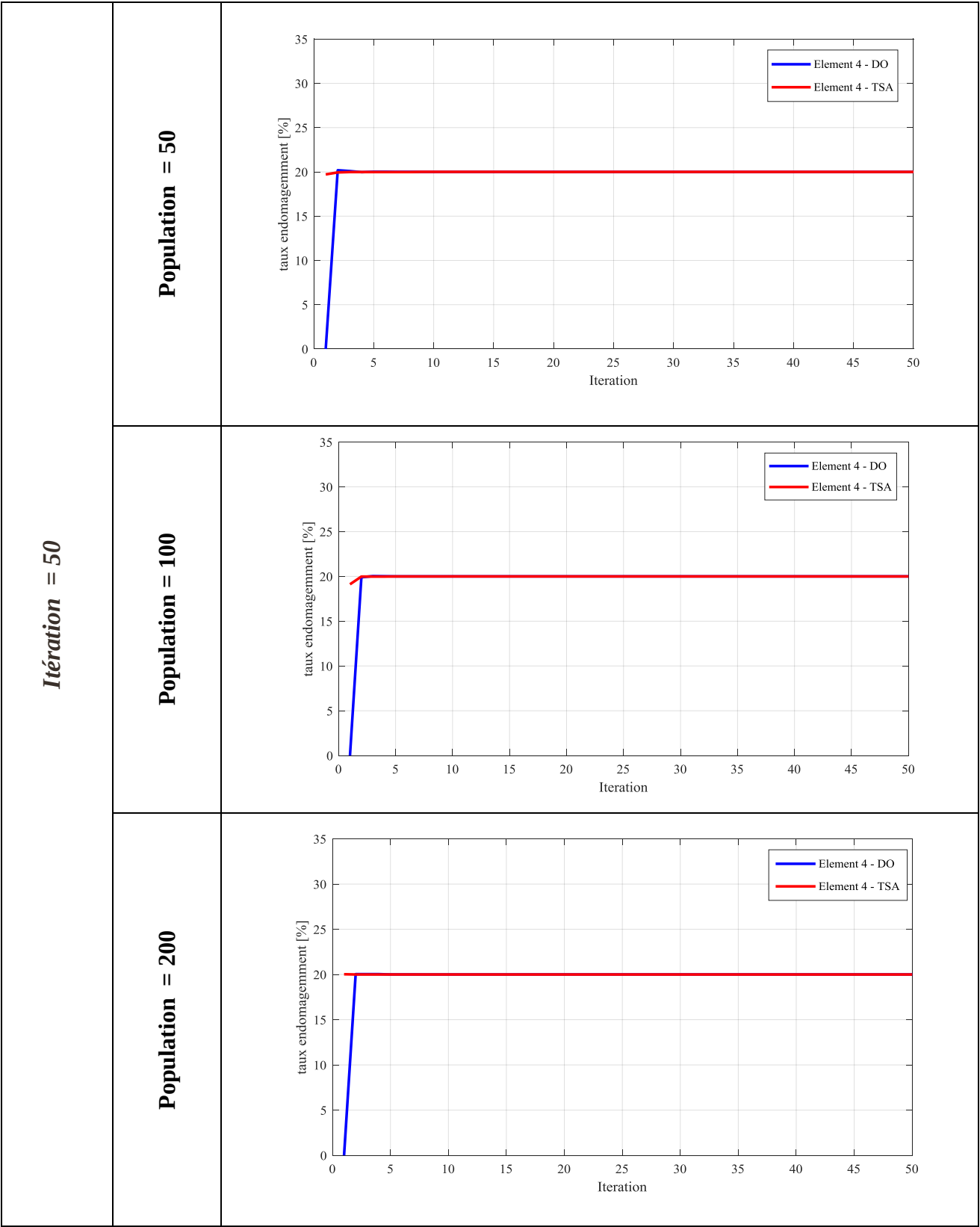


Figure 17 : Convergence des éléments endommagés – cas 1 – barre 2D – population 30.



*Figure 18 : Localisation et quantification des éléments endommagés -cas1-barre2D-
Itération 50*

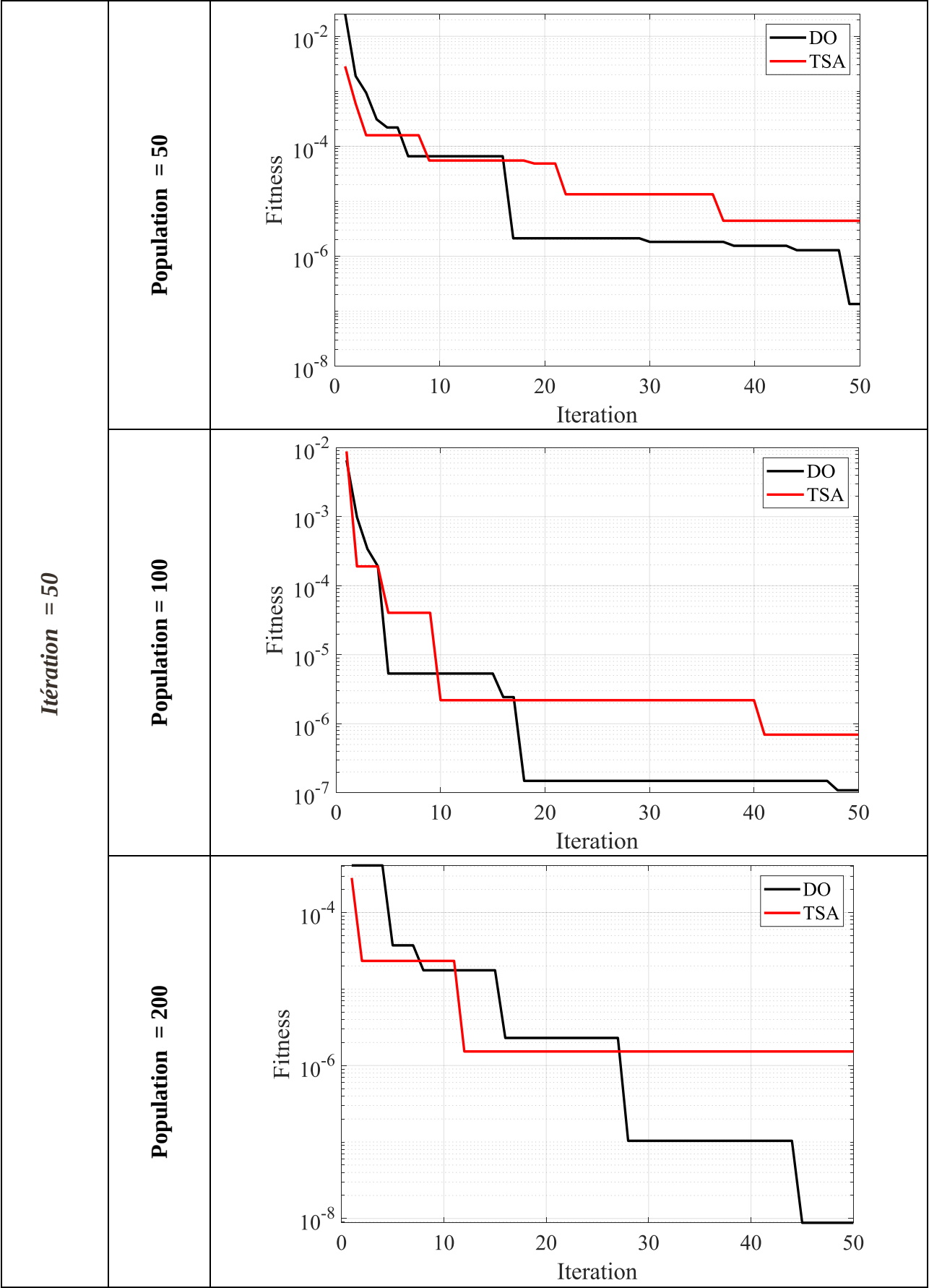


Figure 19 : convergence des éléments –cas1-barre2D-Itération 50

I.6.1.2 Cas 2 deux endommagements :

Pour un nombre de population constant ($P=30$) et un nombre d'itérations qui varie [50,100,200]

Et pour un nombre d'itérations constant ($I=30$) et un nombre de populations qui varie [50,100,200]

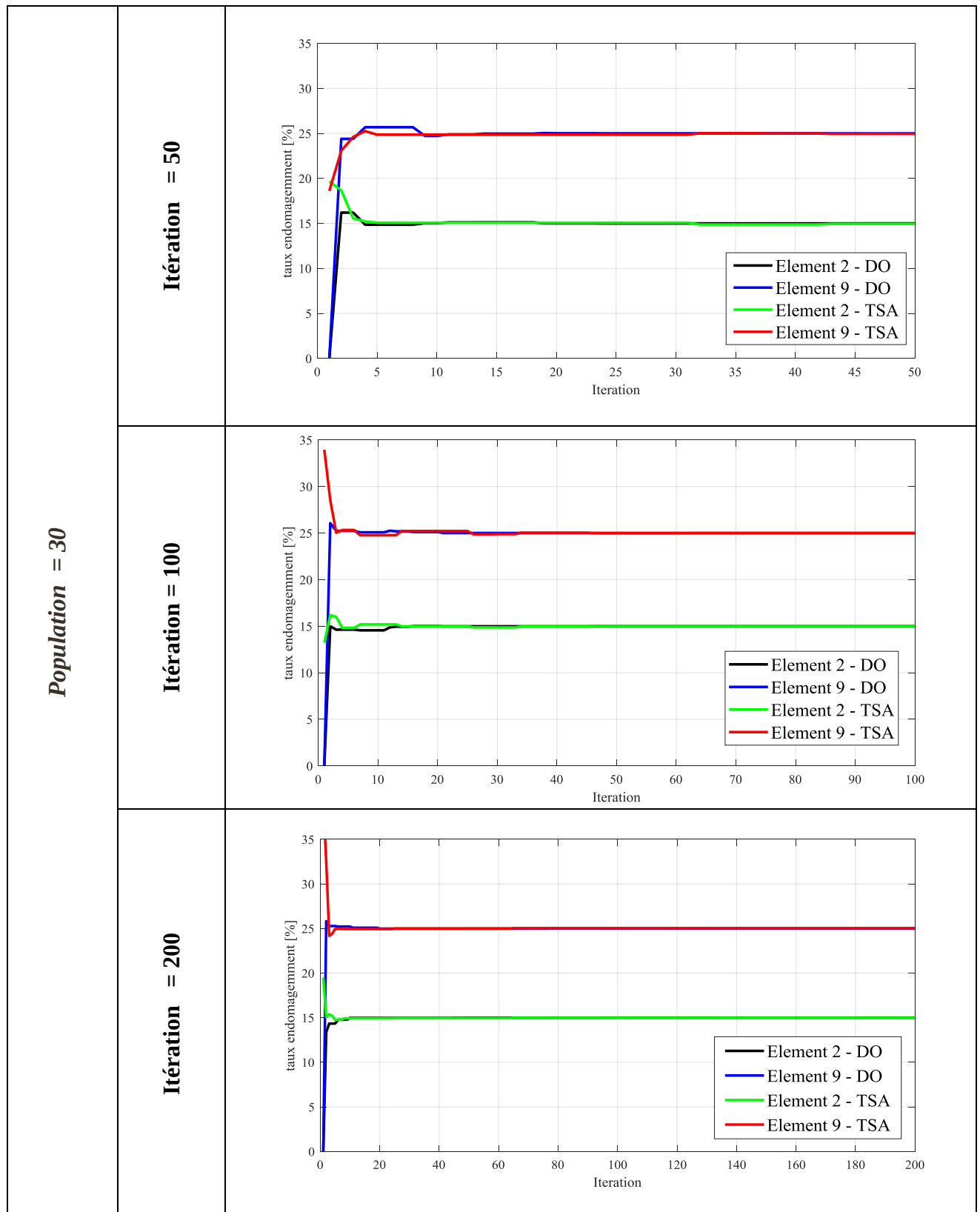


Figure 20 : Localisation et quantification des éléments endommagés –cas2-barre 2D- population 30

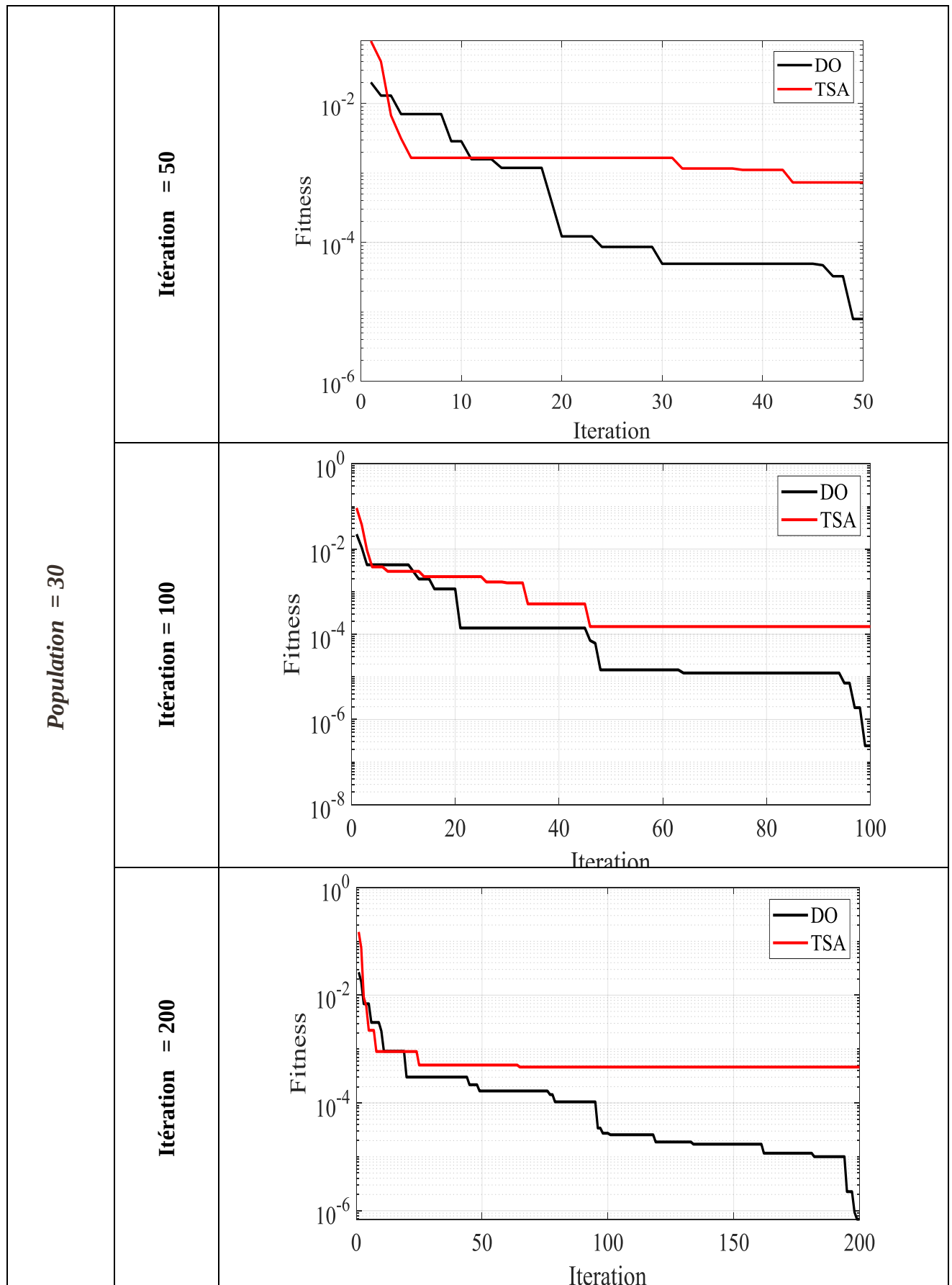


Figure 21: convergence des éléments endommagés –cas2-barre2D-population 30

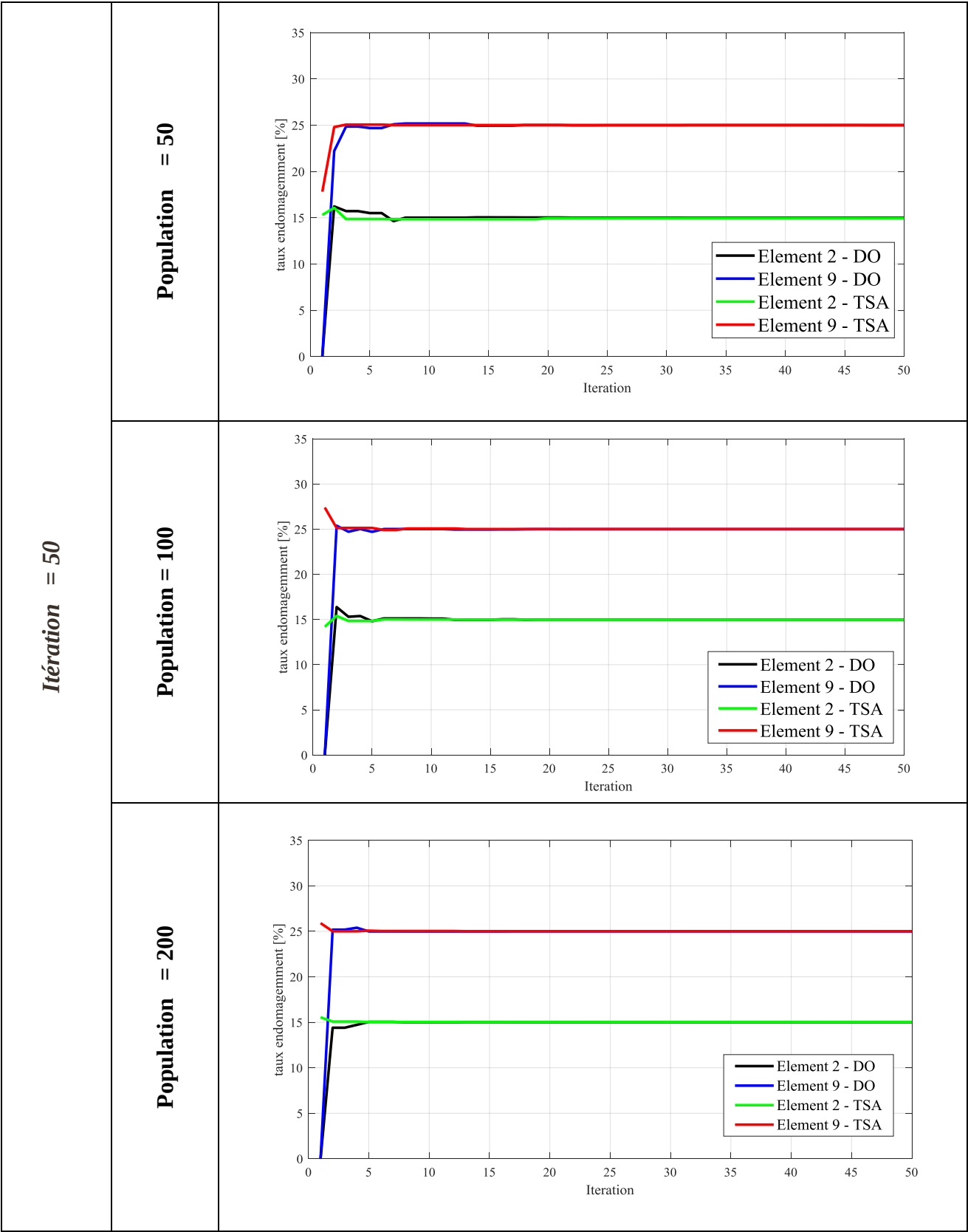


Figure 22: Localisation et quantification des éléments endommagés –Cas 2-Barre 2D-
itération 50

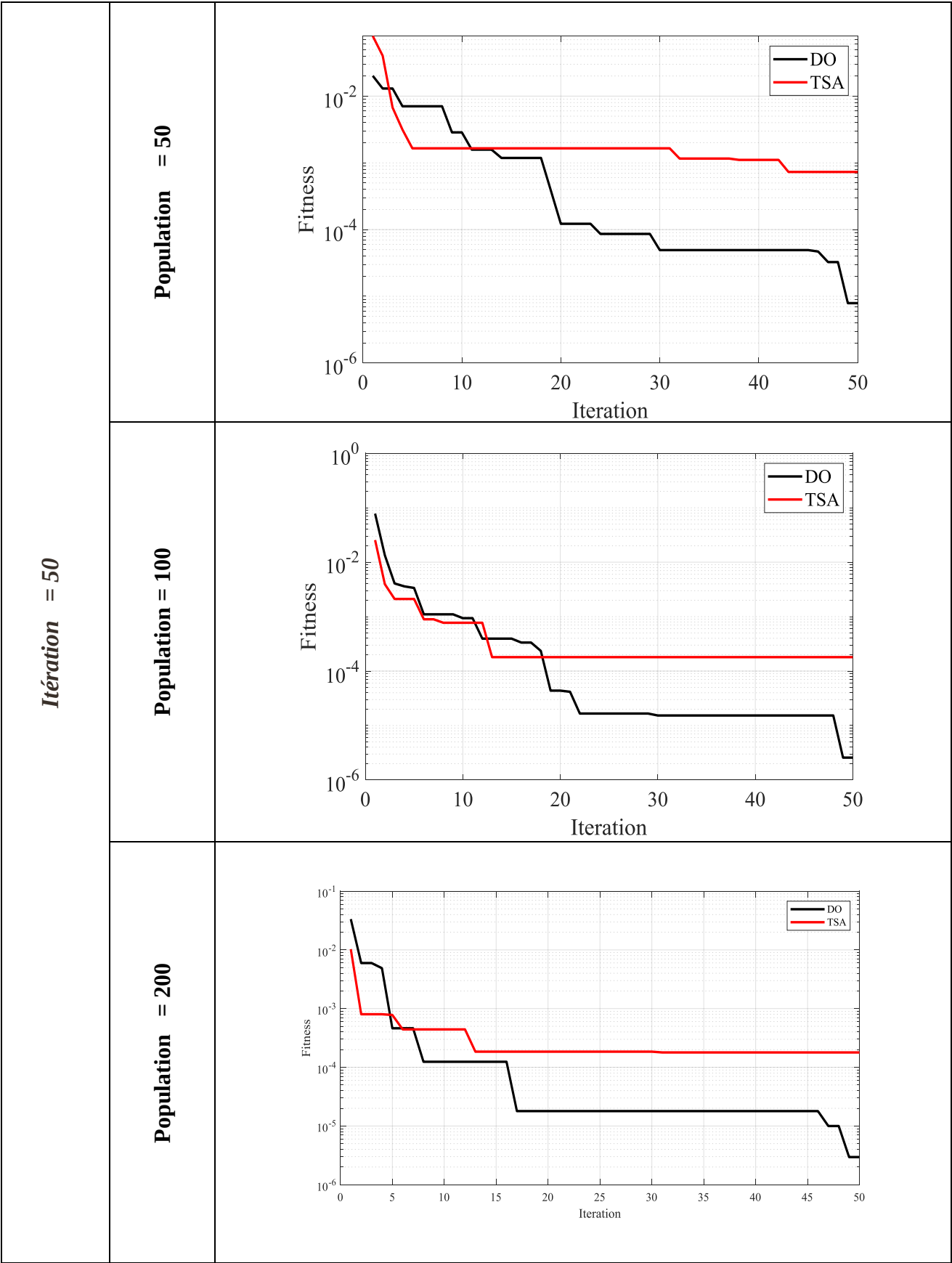


Figure 23:convergence des éléments endommagés –cas 2-barre 2D-iteration 50

I.6.1.3 Cas 3 trois endommagements :

Pour un nombre de population constant ($P=30$) et un nombre d'itérations qui varie [50,100,200]

Et pour un nombre d'itérations constant ($I=30$) et un nombre de populations qui varie [50,100,200]

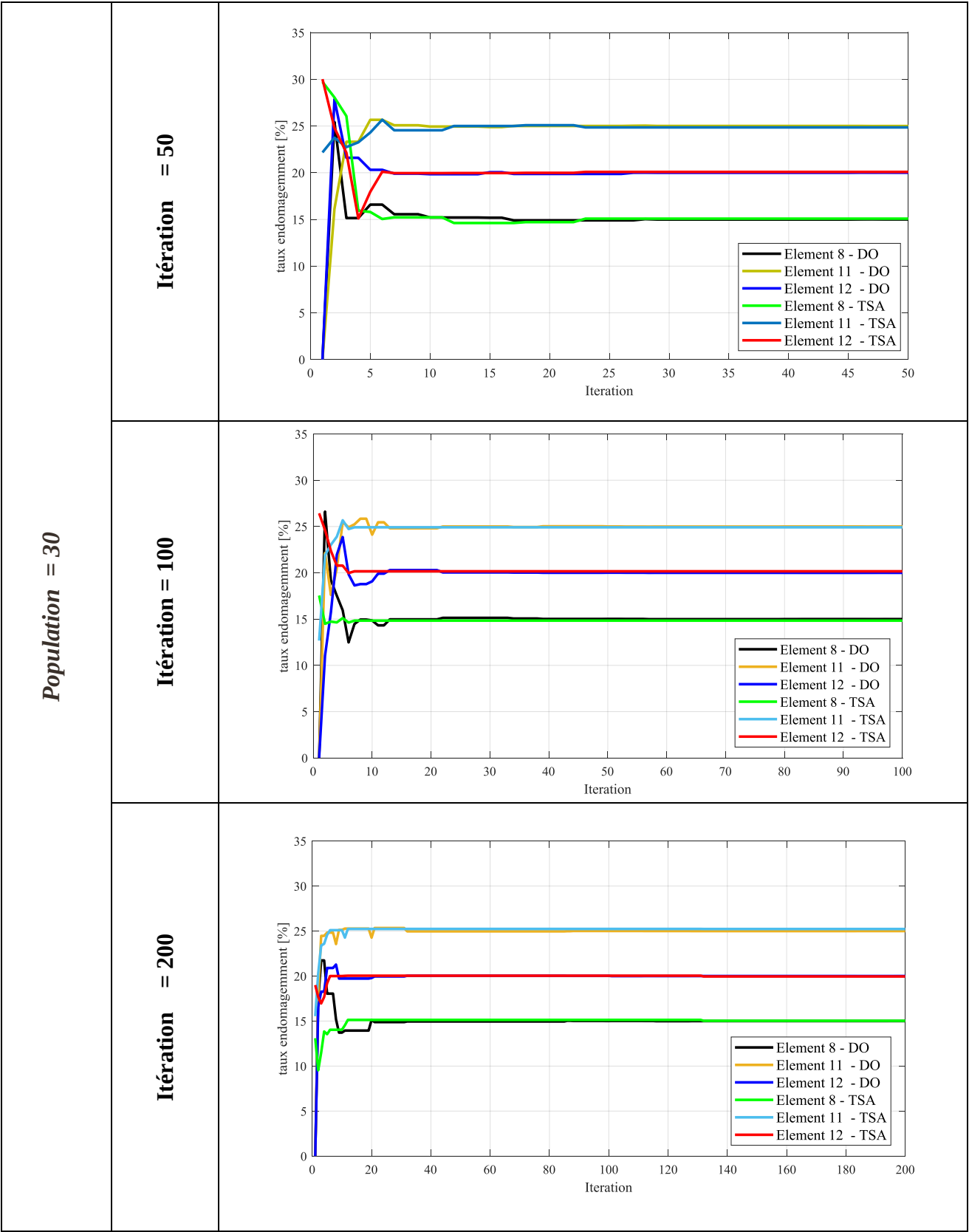


Figure 24 :Localisation et quantification des éléments endommagés –cas3-barre2D- population 30

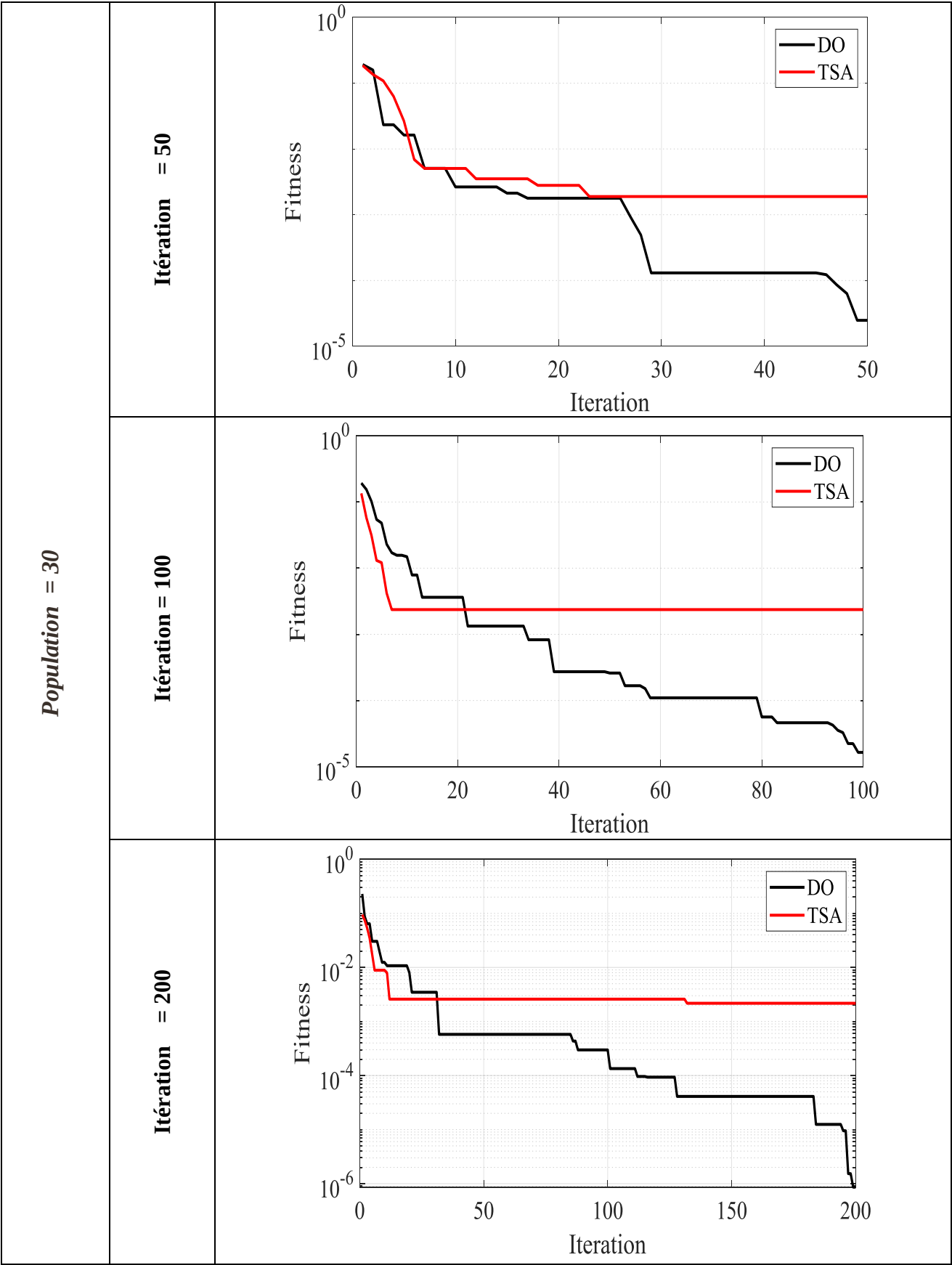


Figure 25 : Convergence des éléments endommagés-cas3- –barre 2 D –population 30

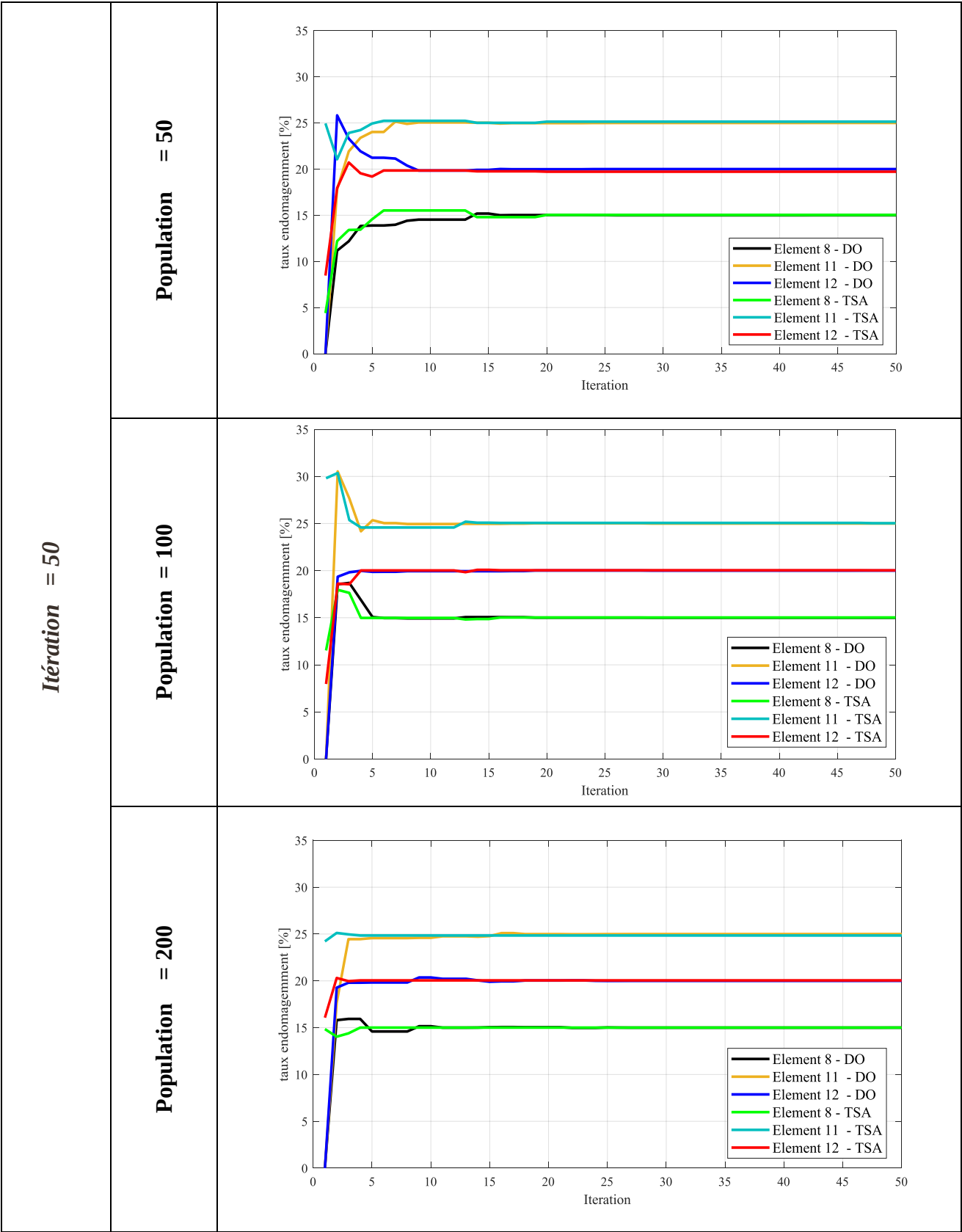


Figure 26 :Localisation et quantification des éléments endommagés –cas3-barre 2D-
iteration 50

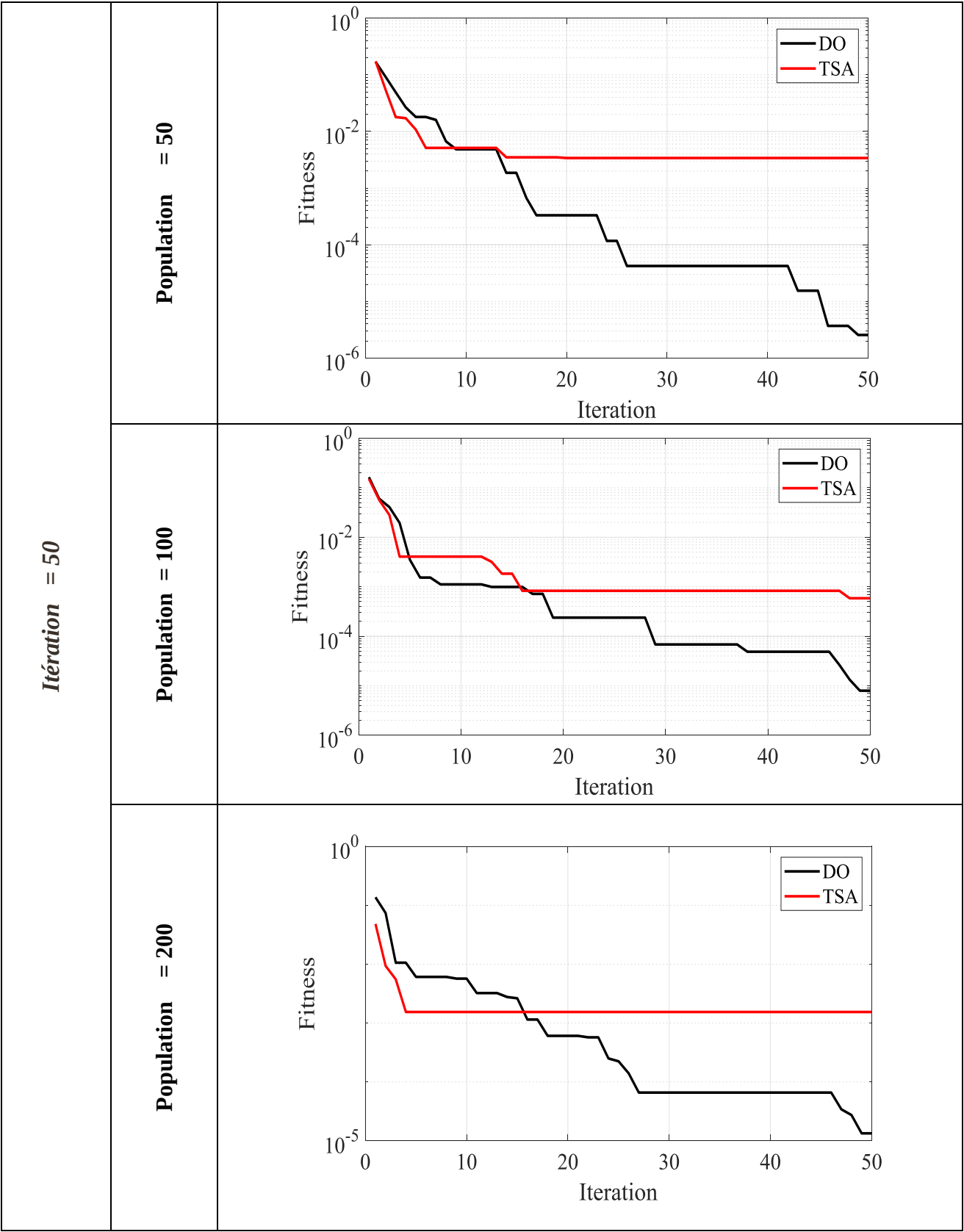


Figure 27 : Convergence des éléments endommages-cas3 –barre 2d –Itération 50

Les graphes obtenus montrent que l'algorithme DO est convergé après peu d'itérations par rapport à l'algorithme TSA et que l'emplacement et la quantification des endommagements (taux %) pour les deux algorithmes sont correctement identifiés.

Les deux algorithmes se convergent plus rapidement pour un nombre d'itérations constante et une variation de la population, mais nous remarquons que l'algorithme DO est plus rapide par rapport à l'algorithme TSA. Cependant, de plus que le nombre d'éléments endommagés augmente, le nombre d'itérations augmente et la convergence devient plus lente mais les taux d'endommagements sont obtenus avec une bonne précision.

I.6.2 Barre 3 D 28 éléments :

Cases		Cas 1				
Population	30			50	100	200
Itération	50	100	200	50		
DO	18,9368179	37,373675	74,6861831	31,2890824	64,0551991	124,927192
TSA	19,8925189	37,6626547	77,7168215	32,8831849	65,0200126	126,262651
Cases		Cas 2				
Population	30			50	100	200
Itération	50	100	200	50		
DO	20,0800476	40,2649429	81,5474239	33,7498709	67,5742786	135,944546
TSA	20,9291989	40,9756113	81,4025387	35,0267531	66,2964595	149,718532
Cases		Cas 3				
Population	30			50	100	200
Itération	50	100	200	50		
DO	10,0753609	19,9793754	40,5781607	17,1628543	32,8740226	67,5809698
TSA	11,3286574	21,5984082	42,2448693	18,232701	34,3256459	69,6881046

Tableau 6. Temps d'exécution des deux algorithmes TSA et DO -Barre3D-

I.6.2.1 Cas 1 un seul endommagement :

Pour un nombre de population constant ($P=30$) et un nombre d'itérations qui varie [50,100,200]

Et pour un nombre d'itérations constant ($I=30$) et un nombre de populations qui varie [50,100,200]

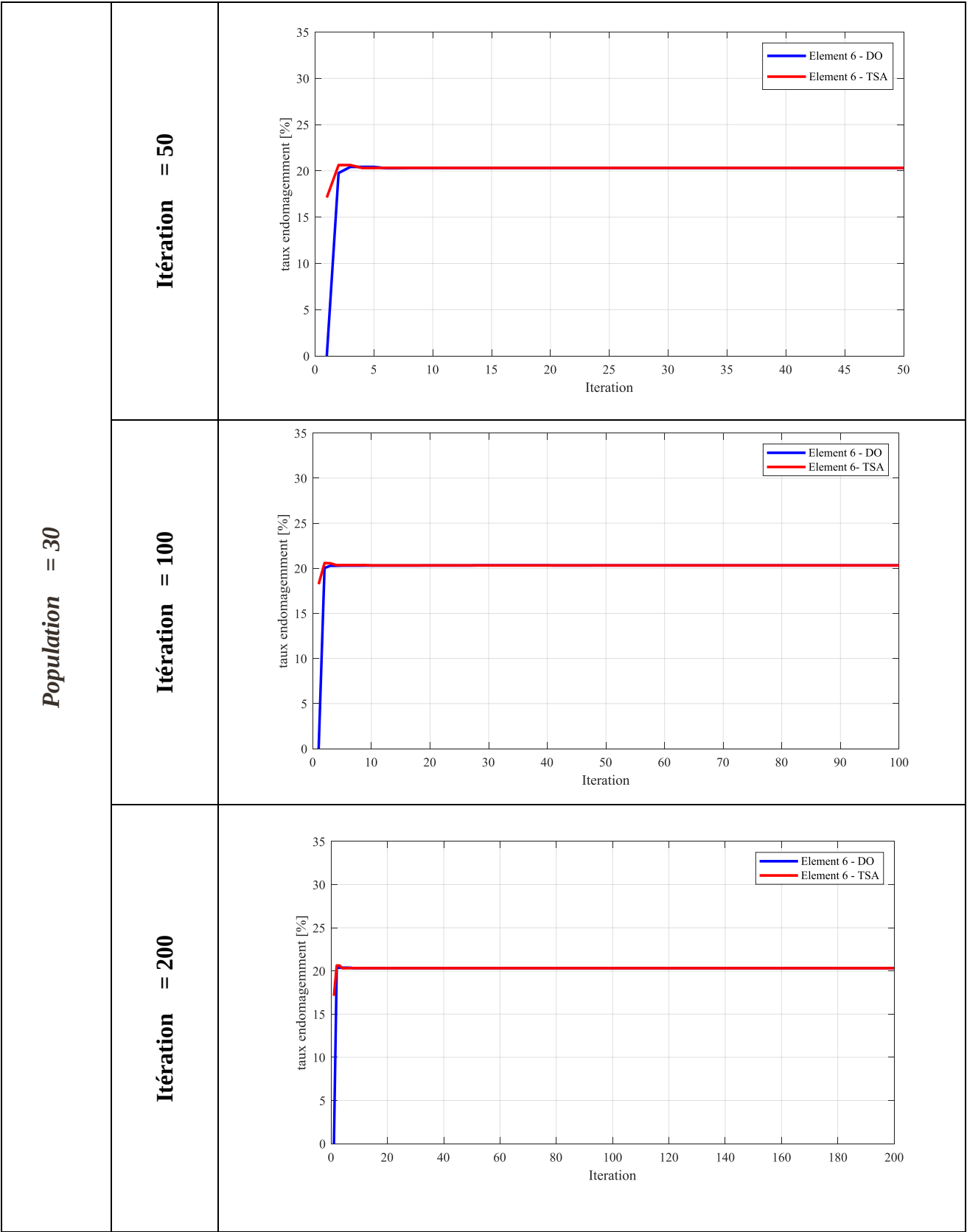


Figure 28:Localisation et quantification des éléments endommagés –cas1-barre3D-
population30

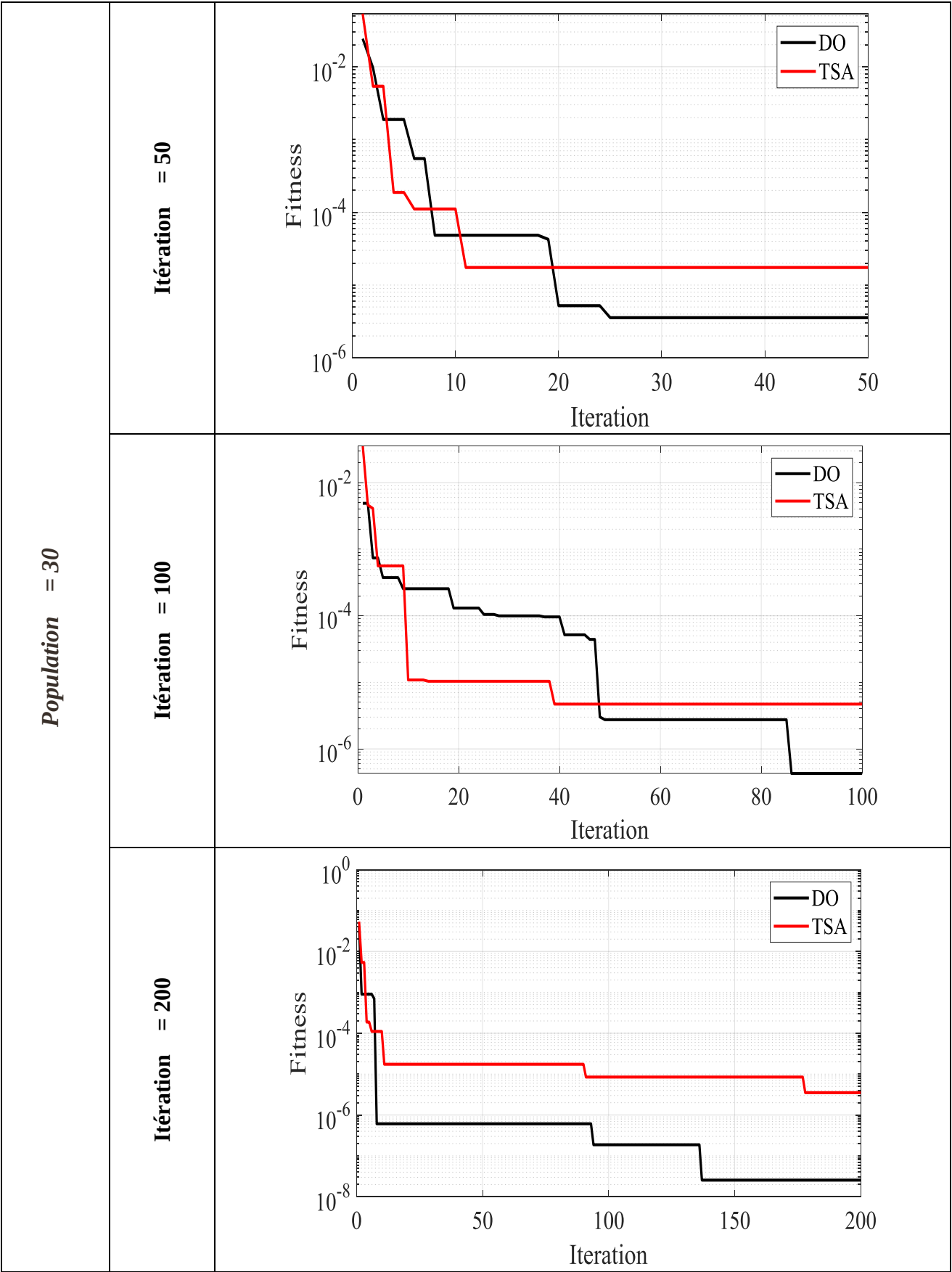


Figure 29: Convergence des éléments endommagés –cas1-barre3D-population 30

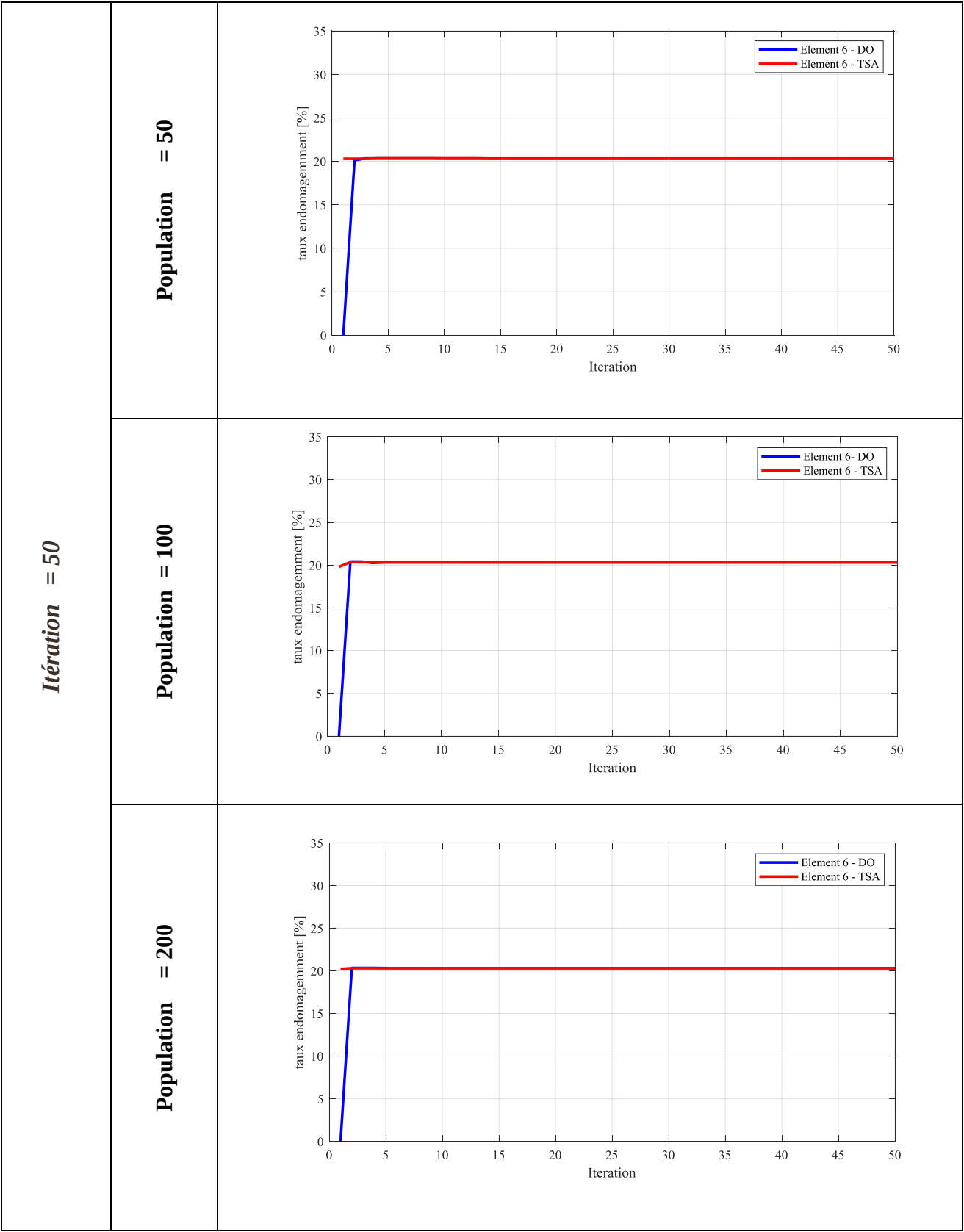


Figure 30 : Localisation et quantification des éléments endommagés –cas1-barre3D-
iteration 50

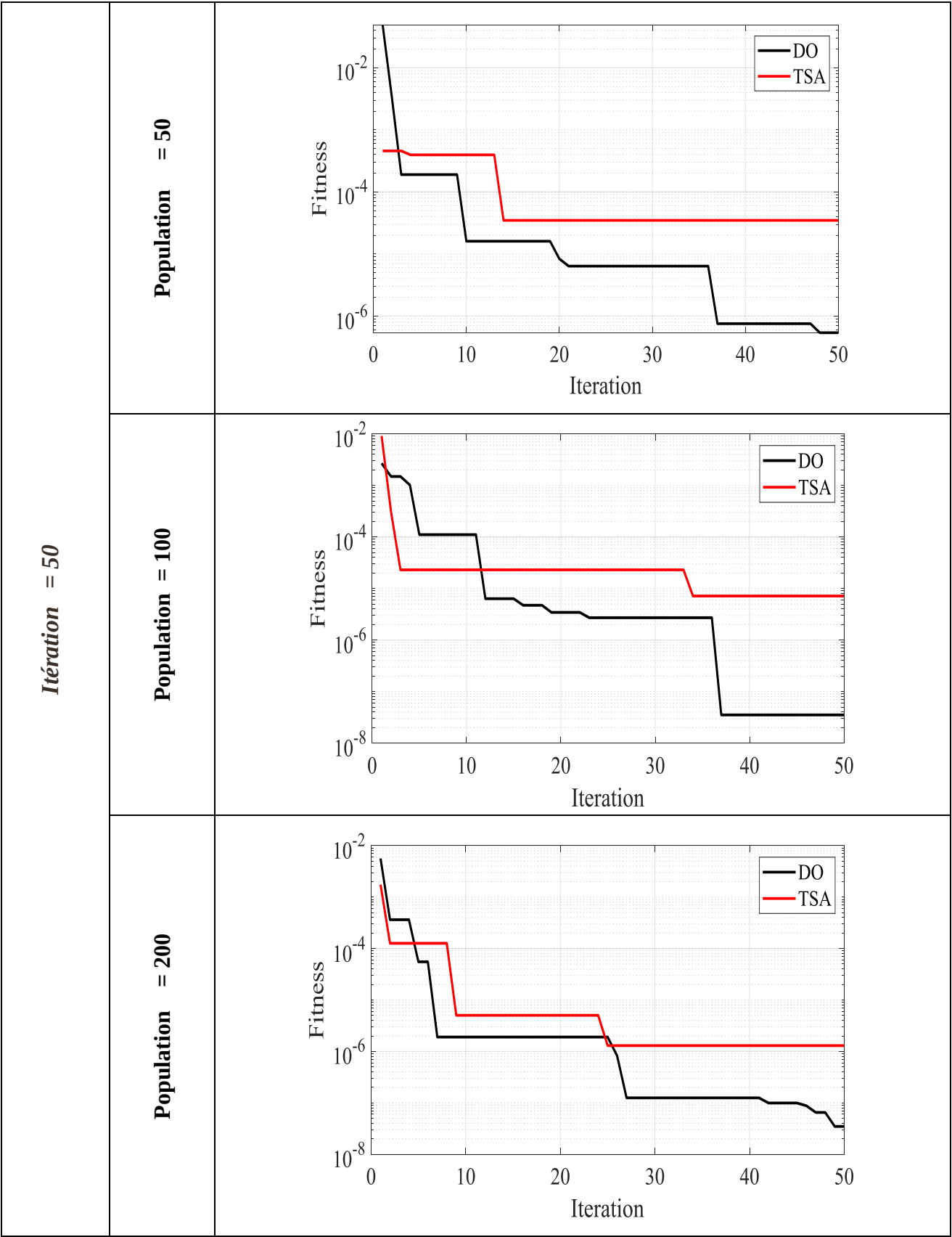


Figure 31 : Figure 30 :convergence des éléments endommagés –Cas 1 –barre 3D- itération 50

I.6.2.2 Cas 2 deux endommagements :

Pour un nombre de population constant ($P=30$) et un nombre d'itérations qui varie [50,100,200]

Et pour un nombre d'itérations constant ($I=30$) et un nombre de populations qui varie [50,100,200]

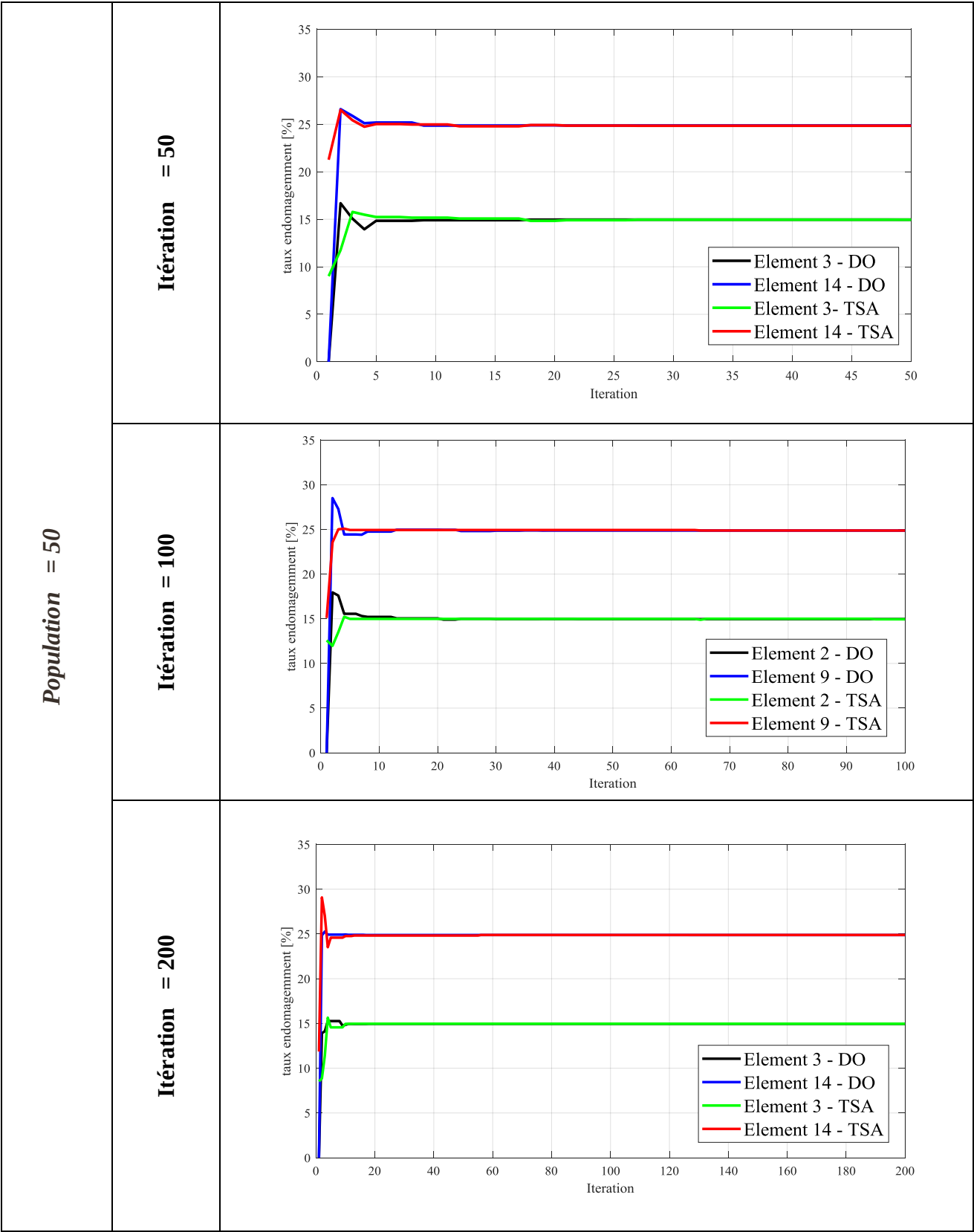


Figure 32 : Localisation et quantification des éléments endommagés –Cas2-Barre3D-
population 30

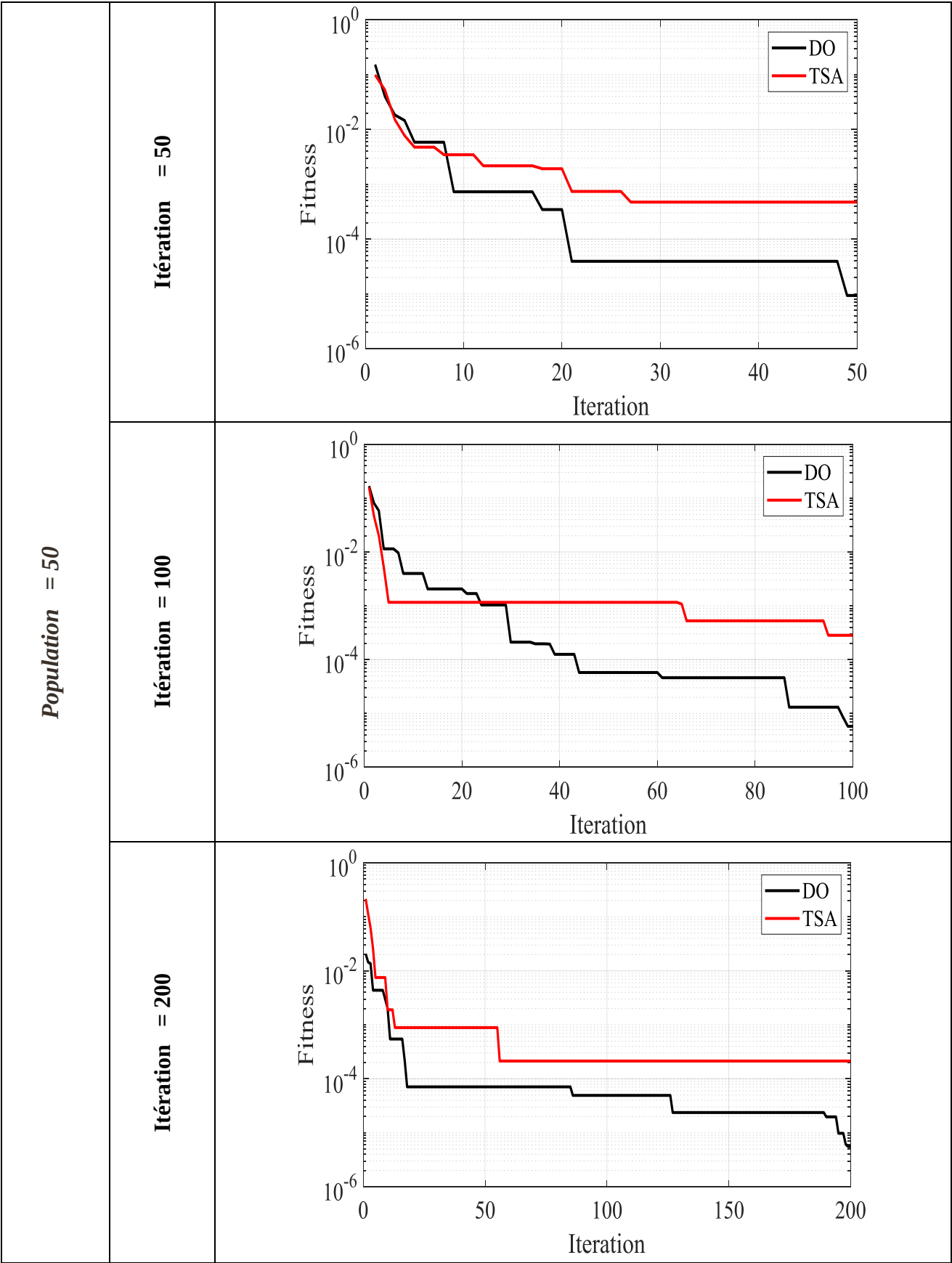


Figure 33 :Convergence des éléments endommagés –Cas2-barre3D-population 30

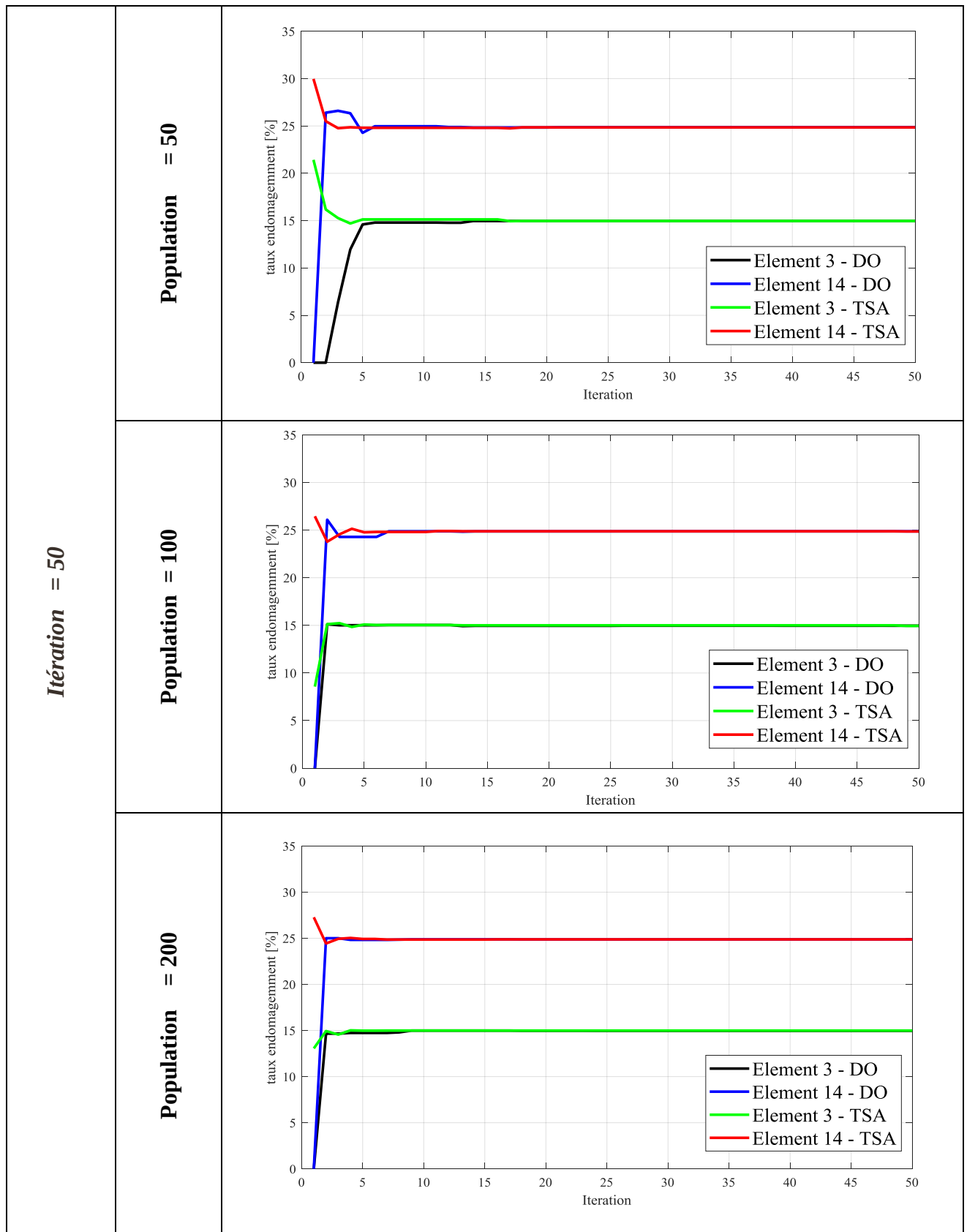


Figure 34: Localisation et quantification des éléments endommagés –cas2-barre3d- itération 50

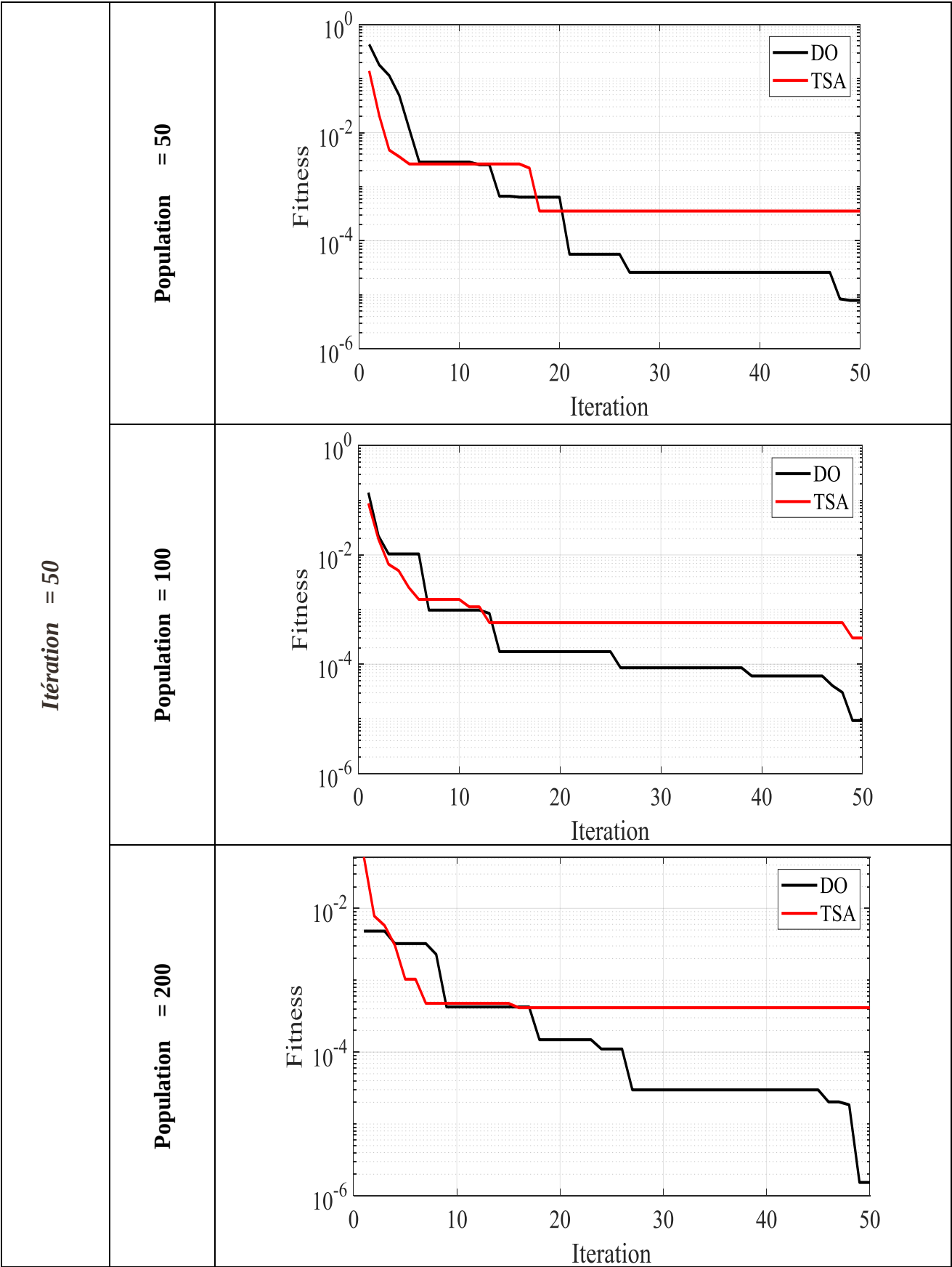


Figure 35:Convergence des éléments endommagés –Cas2-Barre3D-itération50

I.6.2.3 Cas 3 de trois endommagements :

Pour un nombre de population constant ($P=30$) et un nombre d'itérations qui varie [50,100,200]

Et pour un nombre d'itérations constant ($I=30$) et un nombre de populations qui varie [50,100,200]

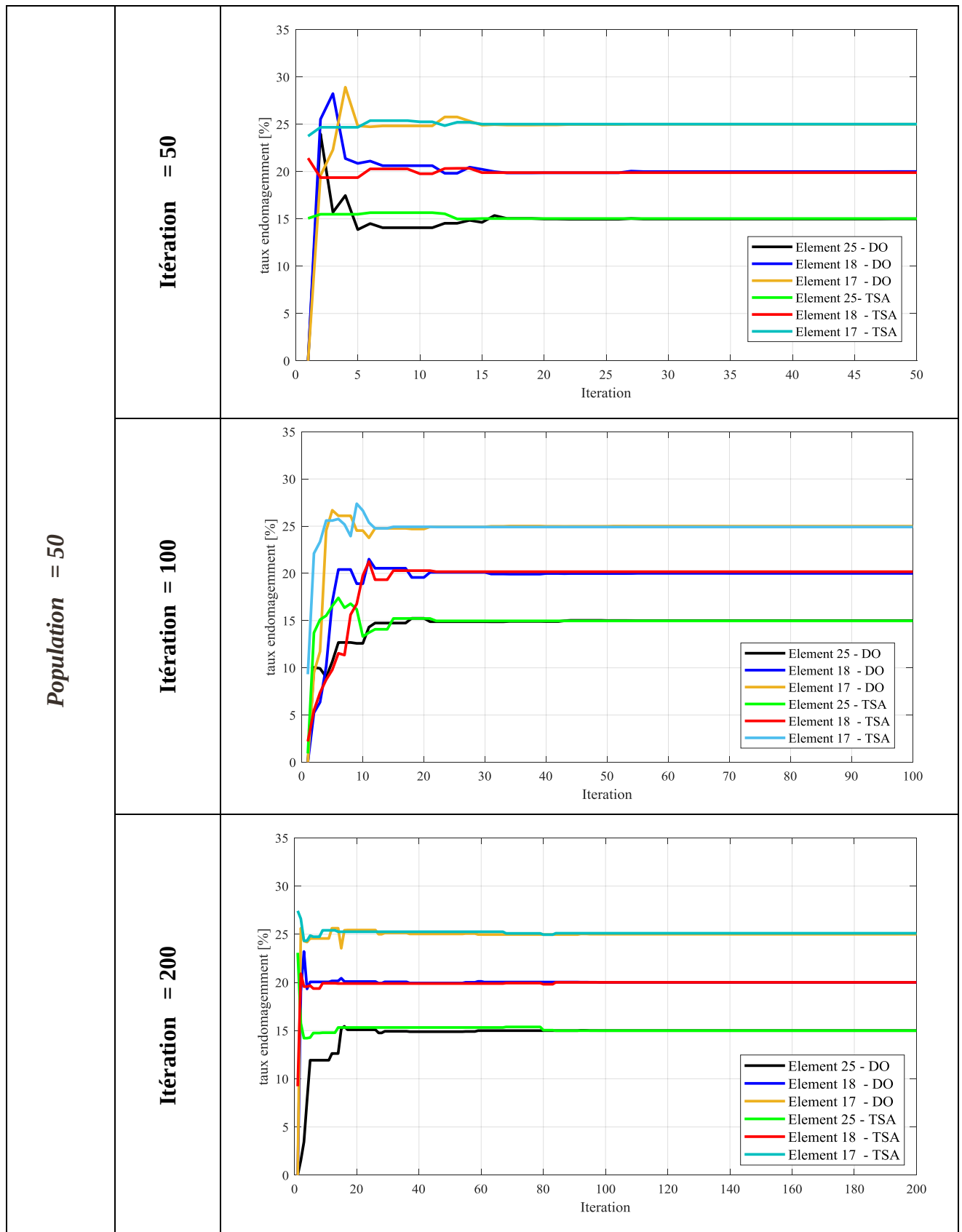


Figure 36: Localisation et quantification des éléments endommagés –Cas3-barre3D- population 30

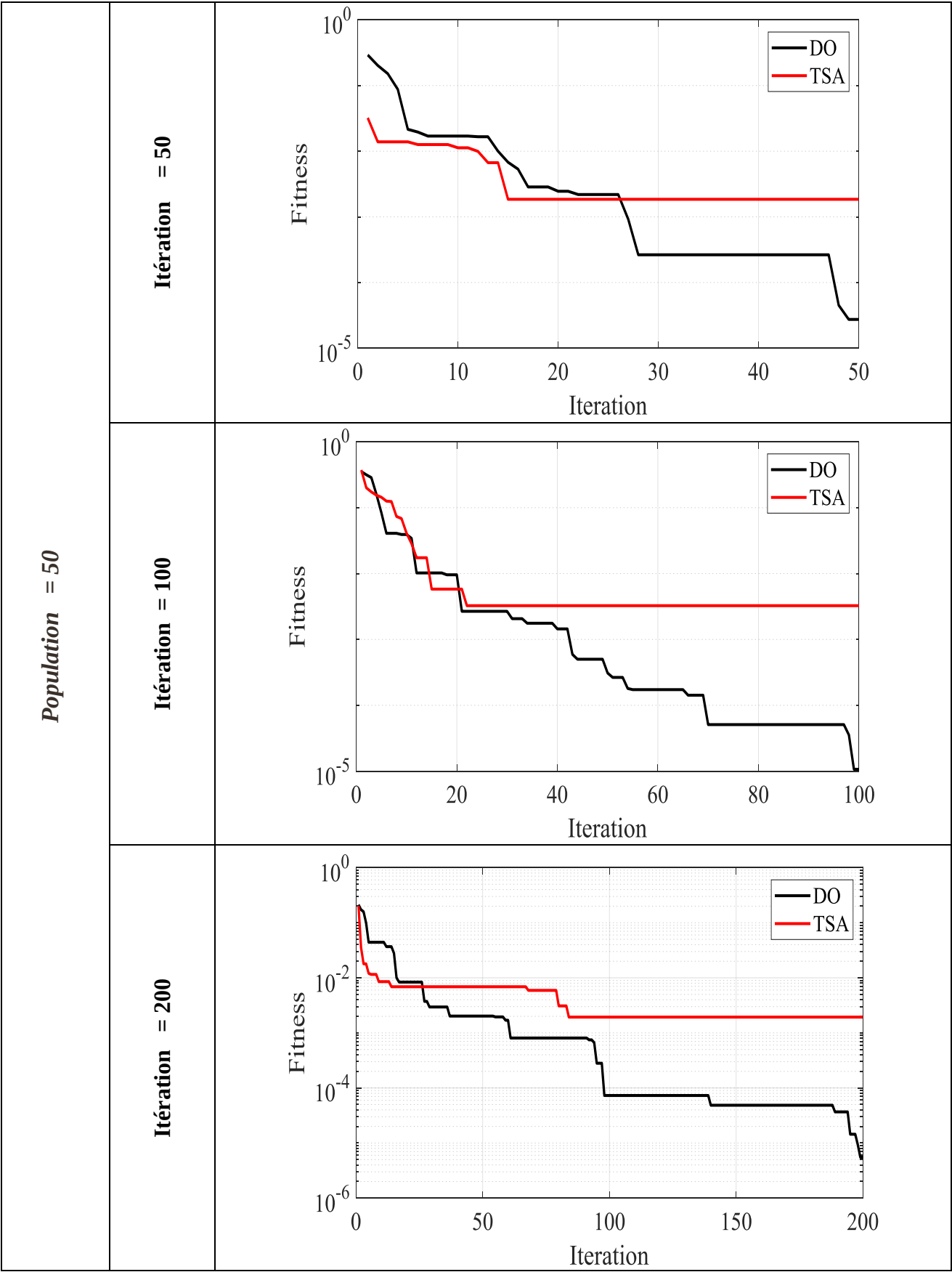


Figure 37:Convergence des éléments endommagés –Cas3-Barre3D-population30

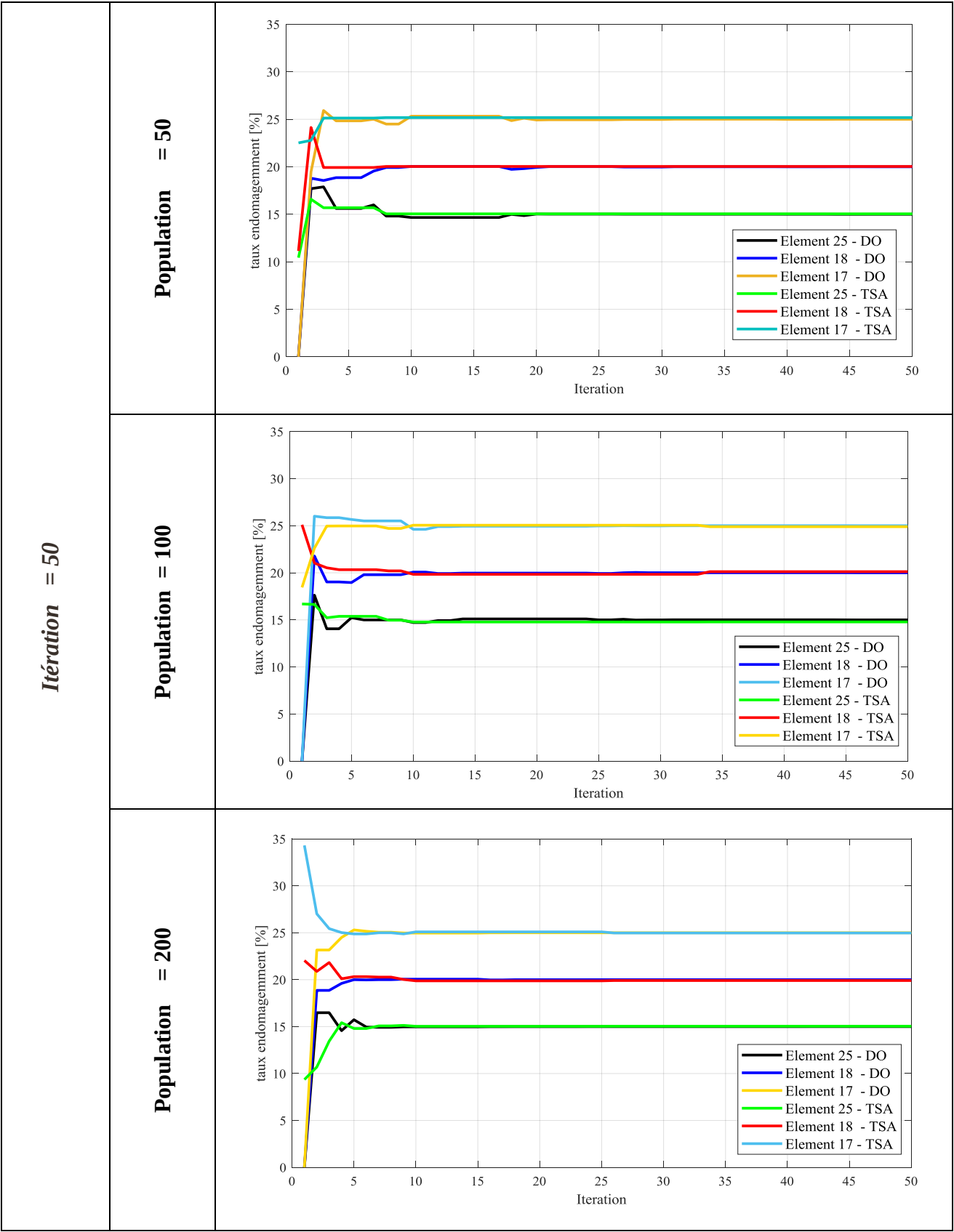


Figure 38:Localisation et quantification des éléments endommagés –Cas3-barre3D- itération 50

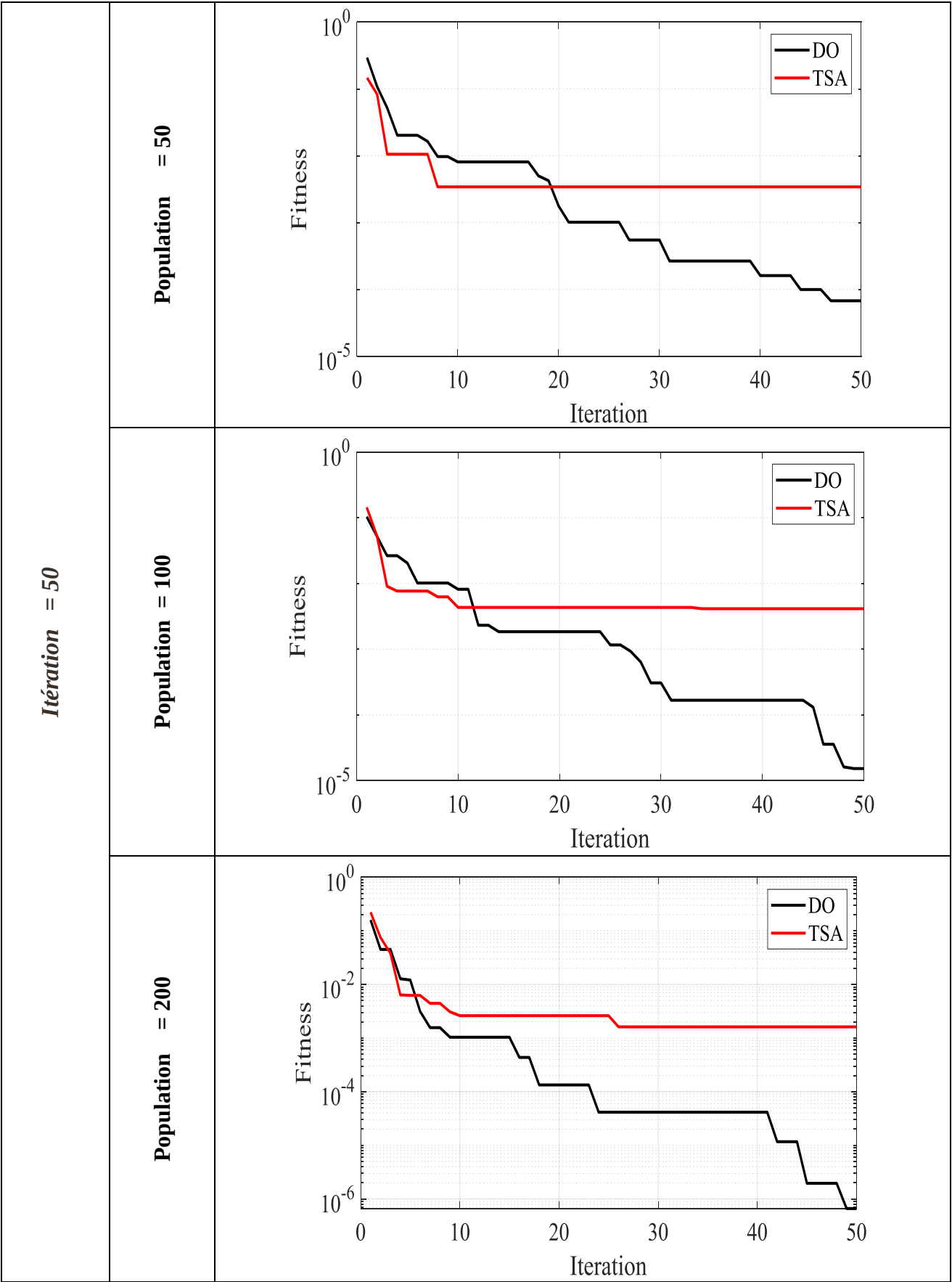


Figure 39:Convergence des éléments endommagés –Cas3-barre3D-itération50

Pour cette structure 3D, nous remarquons la même chose que la structure 2d sauf que la convergence de l'algorithme TSA est vraiment lente car la structure est un peu compliquée.

I.6.3 Conclusion sur les tests :

Pour une meilleure précision et quantification des éléments endommagés il est bien évident d'utiliser l'algorithme DO l'optimiseur de pissenlit d'après les tableaux Tableau 7 Tableau 8 nous remarquons que quel que soit le nombre des éléments endommagés et la complexité de la structure sur laquelle nous appliquerons ces optimiseurs, l'algorithme DO est plus précis et correcte. Pour un meilleur temps d'exécution(un temps court) et une convergence rapide on prend un nombre d'itération qui sera constante en variant les populations, un grand nombre de populations nous permet d'avoir une grande diversité de choix de solution.

L'optimiseur TSA est un algorithme un peu long et pas vraiment précis. C'est à dire il prend du temps pour donner des résultats incorrects pas déterminés. Comme nous avons vu précédemment il est bien mieux d'ajouter à cette algorithme des calculs des méthodes supplémentaires afin d'avoir les meilleurs résultats(ajouter des opérateurs).

Ces deux optimiseurs DO et TSA sont bien utilisés sur des structures simples et pas compliquées.

		Population	30			50	100	200
		Itération	50	100	200	50		
Cas 1	TSA	Elément4	20,00%	20,00%	20,00%	20,00%	20,00%	20,00%
	DO	Elément4	20,00%	20,00%	20,00%	20,00%	20,00%	20,00%
Cas 2	TSA	Elément 2	14,86%	15,02%	14,98%	14,95%	14,99%	15,02%
	TSA	Elément 9	25,03%	25,01%	25,04%	25,01%	25,01%	25,01%
	DO	Elément 2	15,00%	15,00%	15,00%	15,00%	15,00%	15,00%
	DO	Elément 9	25,00%	25,00%	25,00%	25,00%	25,00%	25,00%
Cas 3	TSA	Elément8	15,09%	14,83%	15,03%	15,03%	15,03%	15,01%
	TSA	Elément 12	24,86%	24,92%	25,23%	25,14%	25,03%	24,84%
	TSA	Elément 11	20,09%	20,16%	19,96%	19,72%	20,03%	20,05%
	DO	Elément8	15,00%	15,00%	15,00%	15,00%	15,00%	15,00%
	DO	Elément12	25,00%	25,00%	25,00%	25,00%	25,00%	25,00%
	DO	Element11	20,00%	20,00%	20,00%	20,00%	20,00%	20,00%

Tableau 7. Pourcentage des endommagements localisés -Barre 2D-des deux algorithmes TSA et DO

		Population	30			50	100	200
		Itération	50	100	200	50		
Cas 1	TSA	Elément6	20,32%	20,32%	20,32%	20,32%	20,32%	20,32%
	DO	Elément6	20,32%	20,32%	20,32%	20,32%	20,32%	20,32%
Cas 2	TSA	Elément 3	14,96%	14,94%	14,96%	14,97%	14,95%	14,99%
	TSA	Elément14	24,84%	24,87%	24,88%	24,85%	24,86%	24,88%
	DO	Elément 3	14,96%	14,96%	14,96%	14,96%	14,96%	14,96%
	DO	Elément14	24,87%	24,87%	24,87%	24,87%	24,87%	24,87%
Cas 3	TSA	Elément25	15,00%	14,97%	14,99%	15,05%	14,78%	15,04%
	TSA	Elément18	19,89%	20,18%	20,01%	20,04%	20,12%	19,91%
	TSA	Elément17	25,01%	24,91%	25,10%	25,18%	24,90%	24,98%
	DO	Elément25	15,00%	15,00%	15,00%	15,00%	15,00%	15,00%
	DO	Elément18	20,00%	20,00%	20,00%	20,00%	20,00%	20,00%
	DO	Elément17	25,00%	25,00%	25,00%	25,00%	25,00%	25,00%

Tableau 8. Pourcentage des endommagements localisés -Barre 3D-des deux algorithmes TSA et DO

Conclusion Générale

Comme la technologie est en plein essor avec la disponibilité des tendances émergentes, les méthodes de détection d'endommagements basées sur les caractéristiques vibratoires ont fait l'objet de nombreuses recherches, l'évaluation dynamique est faite par l'identification des paramètres modaux (fréquences propres, les coefficients d'amortissement et les déformées propres) qui sont ensuite utilisé afin d'identifier les endommagements dans des structures.

Dans notre étude nous avons réalisé un système qui comporte deux principales parties, la première concerne la détection des endommagements dans deux structures 2D et 3D en utilisant la nouvelle méthode LFCR basé sur les fréquences locales, la seconde se charge de localiser et quantifier les éléments endommagés.

Nous avons traité la phase de la détection des endommagements par l'une des méthodes de détection qui est basé sur le rapport des changements des fréquences locales en l'appliquant sur deux structures treillis 2D et 3D que nous avons réalisé sur le logiciel Matlab par la méthode des éléments finis ensuite nous avons obtenues les fréquences saines comparés aux fréquences endommagées avec la méthode LFCR nous avons obtenu les éléments endommagés. Et dans la dernière phase de localisation et quantification des éléments endommagés nous avons opté pour des algorithmes d'optimisation ou nous avons utilisé deux sortes d'algorithmes inspiré de la nature sur nos deux structures. Les résultats obtenus ont démontré que l'optimiseur DO est le meilleur algorithme qui quantifie avec une très bonne précision.

Nous estimons avoir réalisé un système répondant à l'objectif que nous avons fixé au départ, à savoir la mise en œuvre d'un système permettant de détecter et localiser et quantifier

les endommagements. En espérant une mise en œuvre d'une amélioration de ce système pour pouvoir l'utiliser dans les besoins de temps réel.

References Bibliographies

1. Bofang Zhu. The finite element method: fundamentals and applications in civil, hydraulic, mechanical and aeronautical engineering by Zhu B. - PDF Drive. <https://www.pdfdrive.com/the-finite-element-method-fundamentals-and-applications-in-civil-hydraulic-mechanical-and-aeronautical-engineering-e189901258.html> (2018).
2. Dimitrios G. Pavlou, PhD. Essentials of the finite element method : for mechanical and structural engineers - PDF Drive. <http://www.pdfdrive.com/essentials-of-the-finite-element-method-for-mechanical-and-structural-engineers-e175823068.html> (2015).
3. M. Paz et al. Structural Dynamics: Theory and Computation - PDF Drive. <http://www.pdfdrive.com/structural-dynamics-theory-and-computation-e187224640.html> (2004).
4. Durrani, A. C A L F E M A FINITE ELEMENT TOOLBOX Version 3.4. (2004).
5. Doebling, S. W., Farrar, C. R. & Prime, M. B. A Summary Review of Vibration-Based Damage Identification Methods. *Identification Methods, " The Shock and Vibration Digest* **30**, 91–105 (1998).
6. lim, chaewoon. Application of vibration-based techniques for modal identification and damage detection in structures. (Concordia University, 2016).

7. A.Güemes et al. J. Compos. Sci. | Free Full-Text | Structural Health Monitoring for Advanced Composite Structures: A Review. <https://www.mdpi.com/2504-477X/4/1/13> (2020).
8. The location of defects in structures from measurements of natural frequencies - P Cawley, R D Adams, 1979. <https://journals.sagepub.com/doi/abs/10.1243/03093247v142049>.
9. Kim, J.-T., Ryu, Y.-S., Cho, H.-M. & Stubbs, N. Damage identification in beam-type structures: frequency-based method vs mode-shape-based method. *Engineering Structures* **25**, 57–67 (2003).
10. Yang, Z. & Wang, L. Structural Damage Detection by Changes in Natural Frequencies. *Journal of Intelligent Material Systems and Structures* **21**, 309–319 (2010).
11. Xu, G. Y., Zhu, W. D. & Emory, B. H. Experimental and Numerical Investigation of Structural Damage Detection Using Changes in Natural Frequencies. *Journal of Vibration and Acoustics* **129**, 686–700 (2007).
12. Xiang, J. & Liang, M. Wavelet-Based Detection of Beam Cracks Using Modal Shape and Frequency Measurements. *Computer-Aided Civil and Infrastructure Engineering* **27**, 439–454 (2012).
13. Samman, M. M. & D, P. Structural Damage Detection Using the Modal Correlation Coefficient (mcc). (1997).
14. Fox, C. H. J. *The location of defects in structures - A comparison of the use of natural frequency and mode shape data*. vol. 1 522–528 (1992).
15. Kim, T. S. & Kim, Y. Y. Mac-based mode-tracking in structural topology optimization.

- Computers & Structures* **74**, 375–383 (2000).
16. Pandey, A. K., Biswas, M. & Samman, M. M. Damage detection from changes in curvature mode shapes. *Journal of Sound and Vibration* **145**, 321–332 (1991).
 17. Curvature mode shape-based damage detection in composite laminated plates - ScienceDirect. <https://www.sciencedirect.com/science/article/abs/pii/S0263822306002376> (2007).
 18. Ciambella, J. & Vestroni, F. The use of modal curvatures for damage localization in beam-type structures. *Journal of Sound and Vibration* **340**, 126–137 (2015).
 19. Pandey, A. K. & Biswas, M. Damage Detection in Structures Using Changes in Flexibility. *Journal of Sound and Vibration* **169**, 3–17 (1994).
 20. Li, J., Wu, B., Zeng, Q. C. & Lim, C. W. A generalized flexibility matrix based approach for structural damage detection. *Journal of Sound and Vibration* **329**, 4583–4587 (2010).
 21. Frizzarin, M., Feng, M. Q., Franchetti, P., Soyoz, S. & Modena, C. Damage detection based on damping analysis of ambient vibration data. *Structural Control and Health Monitoring* **17**, 368–385 (2010).
 22. Kawiecki, G. Modal damping measurement for damage detection. *Smart Mater. Struct.* **10**, 466–471 (2001).
 23. Yang, Q. W. & Liu, J. K. Structural damage identification based on residual force vector. *Journal of Sound and Vibration* **305**, 298–307 (2007).

24. A numerical technique for structural damage detection. *Applied Mathematics and Computation* **215**, 2775–2780 (2009).
25. Shi, Z. Y., Law, S. S. & Zhang, L. M. Structural Damage Detection from Modal Strain Energy Change. *Journal of Engineering Mechanics* **126**, 1216–1223 (2000).
26. Damage Detection Method Based on Element Modal Strain Energy Sensitivity - Wang-Ji Yan, Tian-Li Huang, Wei-Xin Ren, 2010.
<https://journals.sagepub.com/doi/abs/10.1260/1369-4332.13.6.1075>.
27. Min, C.-H., Hong, S., Park, S.-Y. & Park, D.-C. Sensitivity-based finite element model updating with natural frequencies and zero frequencies for damped beam structures. *International Journal of Naval Architecture and Ocean Engineering* **6**, 904–921 (2014).
28. Xu, H. & Humar, J. Damage Detection in a Girder Bridge by Artificial Neural Network Technique. *Computer-Aided Civil and Infrastructure Engineering* **21**, 450–464 (2006).
29. Fang, X., Luo, H. & Tang, J. Structural damage detection using neural network with learning rate improvement. *Computers & Structures* **83**, 2150–2161 (2005).
30. Rhim, J. & Lee, S. W. A neural network approach for damage detection and identification of structures. *Computational Mechanics* **16**, 437–443 (1995).
31. Chou, J.-H. & Ghaboussi, J. Genetic algorithm in structural damage detection. *Computers & Structures* **79**, 1335–1353 (2001).
32. Hao, H. & Xia, Y. Vibration-based Damage Detection of Structures by Genetic Algorithm. *Journal of Computing in Civil Engineering* **16**, 222–229 (2002).

33. Perera, R. & Torres, R. Structural Damage Detection via Modal Data with Genetic Algorithms. *Journal of Structural Engineering* **132**, 1491–1501 (2006).
34. Tiachacht, S. *et al.* Damage assessment in structures using combination of a modified Cornwell indicator and genetic algorithm. *Engineering Structures* **177**, 421–430 (2018).
35. Zenzen, R., Khatir, S., Belaidi, I., Le Thanh, C. & Abdel Wahab, M. A modified transmissibility indicator and Artificial Neural Network for damage identification and quantification in laminated composite structures. *Composite Structures* **248**, 112497 (2020).
36. Radzieński, M., Krawczuk, M. & Palacz, M. Improvement of damage detection methods based on experimental modal parameters. *Mechanical Systems and Signal Processing* **25**, 2169–2190 (2011).
37. Dahak, M., Touat, N. & Benseddiq, N. On the classification of normalized natural frequencies for damage detection in cantilever beam. *Journal of Sound and Vibration* **402**, 70–84 (2017).
38. R. W. Doran « The Gray Code » (2007).
39. Houssein, E. H., Helmy, B. E.-D., Elngar, A. A., Abdelminaam, D. S. & Shaban, H. An Improved Tunicate Swarm Algorithm for Global Optimization and Image Segmentation. *IEEE Access* **9**, 56066–56092 (2021).
40. Kaur, S., Awasthi, L. K., Sangal, A. L. & Dhiman, G. Tunicate Swarm Algorithm: A new bio-inspired based metaheuristic paradigm for global optimization. *Engineering Applications of Artificial Intelligence* **90**, 103541 (2020).
41. Arabali, A., Khajehzadeh, M., Keawsawasvong, S., Mohammed, A. H. & Khan, B. An

- Adaptive Tunicate Swarm Algorithm for Optimization of Shallow Foundation. *IEEE Access* **10**, 39204–39219 (2022).
42. Li, L.-L., Liu, Z.-F., Tseng, M.-L., Zheng, S.-J. & Lim, M. K. Improved tunicate swarm algorithm: Solving the dynamic economic emission dispatch problems. *Applied Soft Computing* **108**, 107504 (2021).
43. Khajehzadeh, M., Iraj, A., Majdi, A., Keawsawasvong, S. & Nehdi, M. L. Adaptive Salp Swarm Algorithm for Optimization of Geotechnical Structures. *Applied Sciences* **12**, 6749 (2022).
44. Li, X., Han, S., Zhao, L., Gong, C. & Liu, X. New Dandelion Algorithm Optimizes Extreme Learning Machine for Biomedical Classification Problems. *Computational Intelligence and Neuroscience* **2017**, e4523754 (2017).
45. Zhu, H., Liu, G., Zhou, M., Xie, Y. & Kang, Q. Dandelion Algorithm With Probability-Based Mutation. *IEEE Access* **7**, 97974–97985 (2019).
46. Han, S., Zhu, K. & Zhou, M. Competition-Driven Dandelion Algorithms With Historical Information Feedback. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **52**, 966–979 (2022).
47. Zhao, S., Zhang, T., Ma, S. & Chen, M. Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. *Engineering Applications of Artificial Intelligence* **114**, 105075 (2022).

