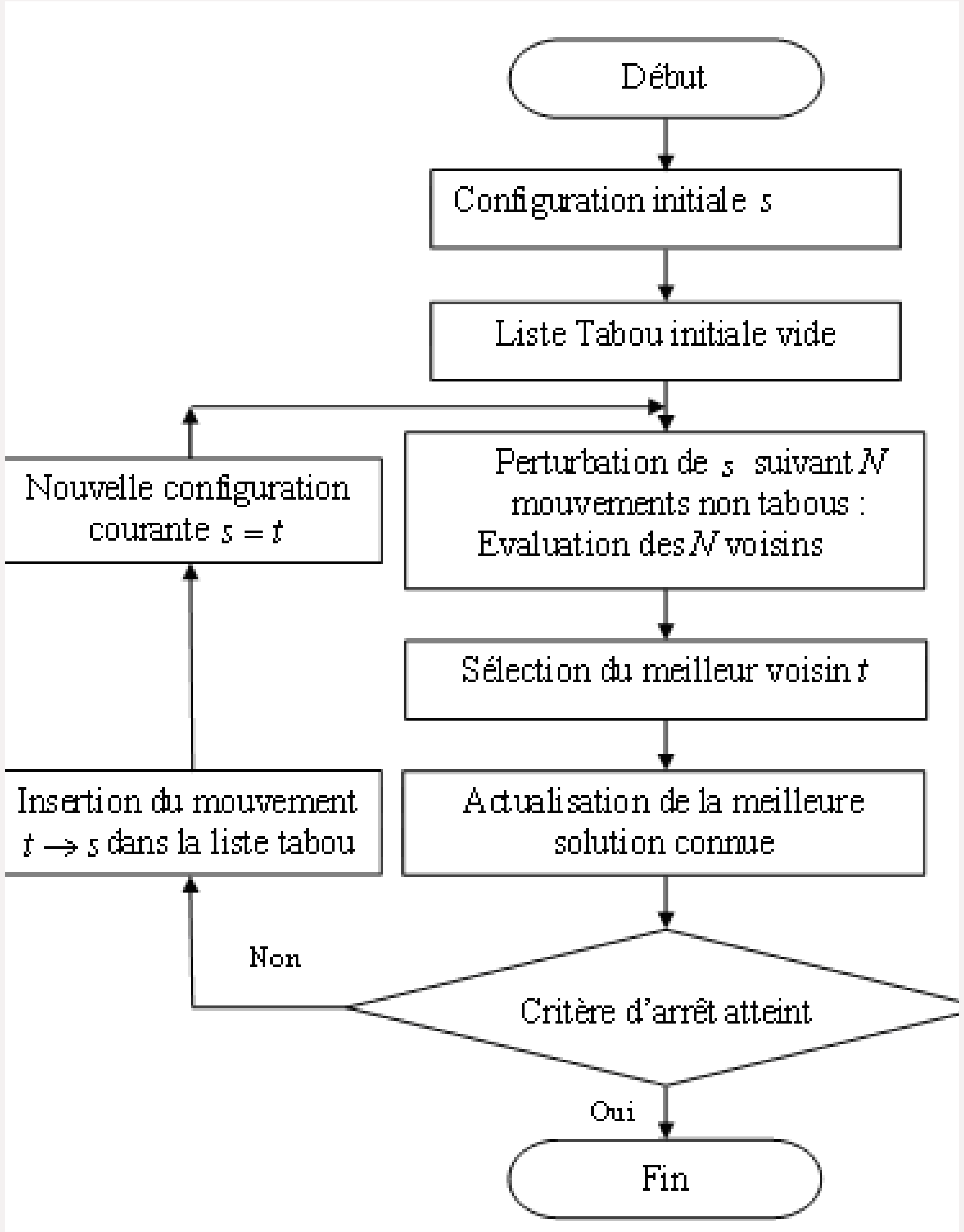
probléme de decoupage

ALGORITHME
DE
TABOU

DÉFINITION

La recherche avec tabou a été proposée par Fred Glover en 1986. Depuis cette date, la méthode est devenue très populaire, grâce aux succès qu'elle a remportés pour résoudre de nombreux problèmes.



concepts de base

- L’IDÉE DE BASE DE LA LISTE TABOUE CONSISTE À MÉMORISER LES CONFIGURATIONS OU RÉGIONS VISITÉES ET À INTRODUIRE DES MÉCANISMES PERMETTANT D’INTERDIRE À LA RECHERCHE DE RETOURNER TROP RAPIDEMENT VERS CES CONFIGURATIONS.
- CES MÉCANISMES SONT DES INTERDICTIONS TEMPORAIRES DE CERTAINS MOUVEMENTS (MOUVEMENTS TABOUS). IL S’AGIT D’INTERDIRE LES MOUVEMENTS QUI RISQUERAIENT D’ANNULER L’EFFET DE MOUVEMENTS EFFECTUÉS RÉCEMMENT (VOIR LES EXEMPLES).
- A CHAQUE ITÉRATION, L’ALGORITHME TABOU CHOISIT LE MEILLEUR VOISIN NON TABOU, MÊME SI CELUI-CI DÉGRADE LA FONCTION DE COÛT. POUR CETTE RAISON, ON DIT DE LA RECHERCHE AVEC TABOU QU’ELLE EST UNE MÉTHODE AGRESSIVE.

pseudo code

Étape 1: choisir une solution initiale i dans S (l’ensemble des solutions)

Appliquer i* = i et k = 0

Étape 2: appliquer k = k+1 et générer un sous-ensemble de solutions en N(i,k) pour que: – les mouvements tabous ne soient pas choisis – un des critères d’aspiration a(i,m) soit applicable

Étape 3: choisir la meilleure solution i’ parmi l’ensemble de solutions voisines N(i,k)

Appliquer i = meilleur i’

Étape 4: si f(i) <= f(i*), alors nous avons trouvé une meilleure solution Appliquer i* = i

Étape 5: mettre à jour la liste T et les critères d’aspiration

Étape 6: si une condition d’arrêt est atteinte, stop. Sinon, retour à Étape 2.

Condition d’arrêt: condition qui régira l’arrêt de l’algorithme

complexité

- xk=xbest=solution initiale
- K=0
- Do
- While(f(xbest)!=0 && k>n)
- For all x’ £N*(xk)
- If f(xk)<f(x*k) then x*=x’
- If f(x’k)<f(xk) then break
- If f(x*k) <f(xbest) then xbest=x*k
- Xk+1=x*k
- Mise a jour liste tabou
- Return xbest

- Il s'agit d’une boucle for dans une boucle

While, qui se répète n fois,

avec n le nombre De voisinage

=>La complexite de cette heuristique est clairement en O(n^2)