
Soumis le : 26 Avril 2010

Forme révisée acceptée le : 09 Juillet 2011

Email de l'auteur correspondant :

m_barr@esi.dz

Approche dirigée par les fourmis pour la fragmentation horizontale dans les entrepôts de données relationnels

BARR Mohamed ^a, BELLATRECHE Ladjel ^b

^a Ecole nationale Supérieure d'Informatique bp 68M Oued Smar, 16270, El-Harrach - Alger - Algérie

^b Ecole Nationale Supérieure de Mécanique et d'Aérotechnique. Laboratoire d'Informatique Scientifique et Industrielle, Université de Poitiers, France

Résumé

Le sujet abordé rentre dans le contexte de l'optimisation des entrepôts de données relationnels. Il consiste à utiliser l'algorithme basé sur les colonies de fourmis pour la sélection de la fragmentation horizontale, qui représente l'une des techniques d'optimisation non redondantes. Le caractère NP-Complet caractérisant le problème de sélection de cette technique justifie le recours aux méthodes approchées ou « méta heuristiques ».

A travers cet article, nous avons modélisé notre problème de sélection d'un schéma de fragmentation horizontale en utilisant l'approche basée sur les colonies de fourmis. Les variables d'entrées sont: l'entrepôt de données non fragmenté, la charge de requêtes fréquemment utilisées et le nombre maximal de fragments exigé par l'administrateur de l'entrepôt de données (AED). Le résultat en sortie est le schéma de fragmentation horizontale qui minimise le coût global de la charge des requêtes. Le paramétrage approprié de dépôt de la phéromone et l'expression significative de l'heuristique qui représente la visibilité, nous ont permis d'obtenir des résultats très satisfaisants. L'expérimentation de notre approche en utilisant un Benchmark (APB1 dans notre cas) est un moyen important pour vérifier l'efficacité de la méthode proposée d'une part, et de pouvoir la situer par rapport à d'autres méthodes qui existent dans ce domaine, d'autre part.

Mots clés : Entrepôt de données, Structures non redondantes, Optimisation, Problème NP-Complet, Métaheuristique, Colonie de fourmis.

1. Introduction et problématique

Les entrepôts de données sont les bases de données destinées à contenir les informations d'aide à la décision. L'obtention des informations d'aide à la prise de décision dans un temps réduit constitue un élément pertinent pour les managers, et représente un challenge pour la sélection d'un schéma de fragmentation horizontale dans les entrepôts de données relationnels.

L'Administrateur de l'Entrepôt de Données (AED) interroge l'entrepôt via des requêtes décisionnelles complexes qui doivent garantir un temps de réponse optimal. Pour réduire le coût d'exécution parfois très important causé par des requêtes décisionnelles complexes invoquant une table de faits très volumineuse et comportant une multitude de jointures entre la table des faits et plusieurs tables de dimensions, l'AED utilise des structures et techniques d'optimisation telles que : les index, les vues matérialisées et/ou la fragmentation de données [3].

Dans la littérature, les problèmes de sélection de ces techniques d'optimisation sont connus NP-complets [3].

Pour résoudre ces problèmes, les chercheurs ont pensé à utiliser les méthodes approchées ou métaheuristiques qui sont capables de trouver une solution quasi-optimale en un temps raisonnable.[6]. Le présent papier se concentre sur le problème de sélection d'un schéma de fragmentation horizontale dans les entrepôts de données relationnels. Ce même problème a fait l'objet d'un travail mené par deux chercheurs en se basant sur l'algorithme génétique [6]. Notre contribution se base sur une approche qui utilise l'algorithme des colonies de fourmis.

2. Organisation de l'article

La suite du papier est organisée comme suit. Dans la section 3, nous commençons par définir la technique de fragmentation horizontale et nous présentons les travaux effectués pour la sélection d'un schéma de fragmentation horizontale dans les entrepôts de données relationnels. Puis, nous montrons la particularité de notre approche par rapport aux approches existantes. La réussite de mapping

entre le problème de sac à dos classique et le problème de sélection d'un schéma de fragmentation horizontale, constitue un point de départ pour modéliser et prendre en charge la fragmentation horizontale. Dans la section 4, nous présentons notre principale contribution : l'algorithme à base des fourmis pour la sélection d'un schéma de fragmentation horizontale. Dans la dernière section, nous présentons notre étude expérimentale en montrant le schéma du benchmark utilisé ainsi que les principaux résultats obtenus.

3. La fragmentation horizontale dans les entrepôts de données relationnels

3.1. Définition

Dans [2], la technique de fragmentation horizontale a été définie comme suit : « La fragmentation horizontale consiste à diviser un ensemble de données en plusieurs partitions, appelées fragments, de manière à ce que la combinaison des fragments recouvre l'intégralité des données sources sans ajout ni perte d'information ».

La fragmentation horizontale peut être déclinée en deux variantes : primaire et dérivée. [5]

- La fragmentation horizontale primaire d'une relation s'effectue grâce à des prédicats simples définis sur un ensemble de données de la même relation.
- La fragmentation horizontale dérivée d'une table consiste à partitionner cette table selon des prédicats définis sur des données d'une autre table [5].

3.2. La complexité de la procédure de fragmentation

Dans [5], il a été démontré que la gestion et la maintenance des fragments horizontaux dérivés est presque impossible si toutes les combinaisons possibles sont retenues.

En effet, si M_i est le nombre de fragments de la table de dimension D_i , et K le nombre de tables de dimensions fragmentées, alors le nombre total de fragments de la table de faits s'élève à :

$$N = \prod_{i=1}^k M_i \quad [6].$$

A ce titre, nous rappelons que les techniques à base de construction de prédicats et à base d'affinités ne prennent pas en charge le phénomène d'explosion du nombre de fragments. [2]

Par contre les approches basées sur la fouille de données qui exploitent k-means peuvent limiter le nombre important de fragments [2].

Dans [6], les auteurs ont utilisé l'algorithme génétique et ils ont tenu compte dans leur travail du problème d'explosion du nombre de fragments.

3.3. Approches de fragmentation horizontale

Dans la littérature, nous trouvons plusieurs approches utilisées pour sélectionner un schéma de fragmentation horizontale à savoir : (1) les approches basées sur la construction des prédicats, (2) les approches basées sur les affinités, et (3) les approches basées sur la fouille de données. [2]

Approches basées sur les prédicats : Le principe de ces approches revient à identifier les sous-ensembles des prédicats contenus dans les tables de dimensions à condition que ces prédicats garantissent la minimalité (disjonction des fragments obtenus deux à deux) et la complétude (possibilité de la reconstruction d'une relation à l'aide de l'union de l'ensemble des fragments la constituant). Ces sous ensembles sont exploités ensuite pour la fragmentation dérivée. [2]

Approches basées sur l'affinité : En outre des prédicats simples de sélection, ces approches se basent sur l'utilisation des fréquences des requêtes de la charge pour construire les fragments des dimensions.

Approches basées sur la fouille de données : Ces approches exploitent les algorithmes de fouilles de données et l'analyse intelligente des données pour sélectionner un schéma de fragmentation [2].

Dans [2], l'auteur montre que son approche a fait ses preuves pour la sélection des structures de données aidant à améliorer les performances d'un système de gestion de bases de données. Dans le contexte de la technique de fragmentation horizontale, plusieurs travaux ont été développés pour la prendre en charge en se basant sur la fouille de données [2]. Nous pouvons citer aussi l'approche de classification par l'algorithme k-means qui vise à obtenir des fragments répondant à des requêtes représentant des caractéristiques communes [2].

4. Notre approche

L'approche que nous avons adoptée est une démarche à base de coût. Elle se distingue par rapport aux approches existantes par le fait que ces dernières procèdent par l'application directe d'un algorithme de résolution du problème de fragmentation horizontale. Dans notre démarche, nous avons rajouté une nouvelle brique qui consiste à mapper le problème de sélection de la technique de fragmentation horizontale à un problème qui existe déjà et qui a fait l'objet de plusieurs sujets de recherche. D'où la possibilité d'exploiter tous les travaux de recherche menés pour résoudre le problème mappé afin d'aboutir à des résultats satisfaisants. La figure 1 schématise notre approche.

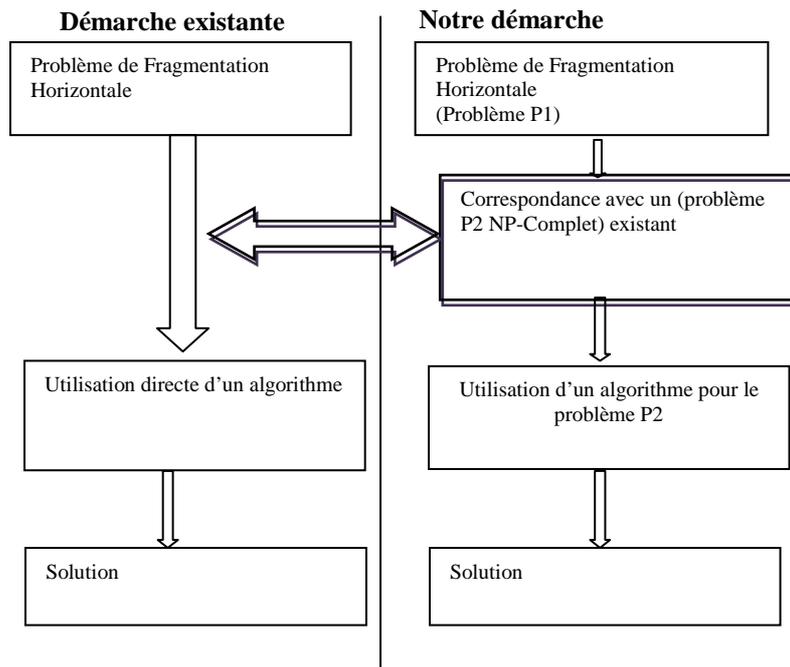


Figure 1 Notre approche

4.1. Correspondance du problème de fragmentation horizontale à un problème de sac à dos

Le problème de sac à dos représente l'un des problèmes NP-Complets classiques. Il consiste à remplir dans un sac un nombre d'objets tels que le poids total des objets ne dépasse pas le poids supporté par le sac, à condition que chaque objet doive apporter un profit. [4]

Table 1

Correspondance entre le problème de sac à dos classique et la fragmentation horizontale

Sac A Dos	Fragmentation Horizontale
Taille du sac : Nombre d'objets à mettre	Nombre de fragments exigé par le DBA, soit $W = 2^n$ (suivant le modèle ÖZSU) Avec n le nombre de prédicats introduits dans le schéma de fragmentation
Profit d'un objet O_i : p_i	Profit du prédicat de sélection = Nombre d'E/S (table de faits) – Nombre d' E/S(fragment introduisant le prédicat)
Poids d'un objet	Charge du prédicat ou nombre d'E/S d'une requête qui inclut ce prédicat
Fonction objectif : Maximiser	
La fonction valeur totale	Maximiser :
	Coût $(Q, SFi) = (E/S(Q, EDNF) - E/S(Q, SFi))$
$X(Z) = \sum_{i=1}^n x_i * p_i$	EDNF: Entrepôt de données non fragmenté
$x_i \in \{0, 1\}$	SFi: schéma de fragmentation

4.2. Un nouveau sens du poids

Du moment que la technique de fragmentation horizontale n'exige pas d'espace supplémentaire tel que le

Dans la littérature, les chercheurs qui ont assimilé le problème de sélection des deux structures d'optimisation à savoir les index et les vues matérialisées, à un problème de sac à dos, sont partis, de la contrainte unique de l'espace de stockage à ne pas dépasser. Cette capacité de stockage représente le poids à respecter et qui ne doit pas dépasser le poids total du sac à dos [3]. Le tableau 1 présente la correspondance que nous avons identifiée entre le problème du sac à dos et celui de fragmentation horizontale.

cas des structures redondantes des index et des vues matérialisées, alors nous avons pensé à une autre charge ou poids pour formaliser le problème de la fragmentation horizontale selon le problème de sac à dos. Pour cela, nous avons pris la quantité de vas et viens entre disque et

mémoire centrale comme poids des objets à mettre dans le sac. Dans le même contexte nous avons considéré les prédicats qui vont participer au schéma de fragmentation comme objets à mettre dans le sac.

En effet, si une requête Q qui introduit un prédicat simple de sélection P¹ de la manière suivante :

```
Select Time_level, count (*)
From ACTVARS A, PRODLEVEL P
Where
A.PRODUCT_LEVEL=P.CODE_LEVEL and
P.Class_Level='A1DGFSPJTJ473'
Group by Time_level
```

Le poids lié au prédicat P, est égal au nombre de pages disque nécessaire pour ramener toutes les lignes de la table des faits et qui impliquent le prédicat simple de sélection introduit dans la requête.

4.3. Estimation du poids

Dans [6], la formule qui estime le nombre de pages disque pour supporter un fragment relatif à un prédicat Pi (Coût (Q, SFi)) introduit dans une requête Q de la charge, est exprimée comme suit :

Coût (Q, SFi) = Fréquence(Q) * (Sél(Pi) * ||F|| * L)/PS
 (Sél(Pi) , ||F||, L, et PS représentent respectivement la Sélectivité de Pi, la taille de la table des faits, la longueur d'un enregistrement de la table des faits, et la taille d'une page disque) ;

4.4. Expression du profit

Dans la littérature, le profit apporté par la structure d'optimisation de l'index par exemple s'exprime par la différence entre le temps de réponse d'une requête sans indexation et le temps de réponse de la même requête en introduisant l'index. [3]

Pour la technique de fragmentation horizontale que nous somme en train d'optimiser, nous avons pris la différence en nombre d'Entrées/Sorties d'une requête avant et après la fragmentation comme valeur de profit (ou utilité).

4.5. Architecture générale de notre solution

Pour mieux comprendre tout le processus de notre solution, une architecture générale de cette dernière est illustrée à travers la figure 2.

Le schéma de la figure 2 met en évidence en entrées la charge des requêtes à partir de laquelle nous avons extrait tous les prédicats simples de partitionnement. (Etape 1 dans la figure).

W représente le nombre de fragments exigé par l'Administrateur de l'entrepôt de données est un élément

hypothétique très important dans le formalisme que nous avons utilisé. (Etape 2 dans la figure).

En fonction de W, et les charges des prédicats au sens de nombre d'Entrées/Sorties entre mémoire et disque qui représente la variable d'optimisation de notre fonction objectif par rapport à un modèle de coût, les fourmis vont sélectionner les bons prédicats regroupés éventuellement en sous ensembles de prédicats de même attribut. (Etape 3 dans la figure).

Les sous ensembles de prédicats sélectionnés seront ceux retenus suivant le modèle ÖZSU² en conjonction de prédicats, pour partitionner l'Entrepôt de Données. (Etape 4 dans la figure). Un schéma de partitionnement final sera déduit. (Etape 5 dans la figure).

Le sac à dos par les fourmis : Dans [7], les auteurs ont traité un travail qui prend en charge la résolution du problème de sac à dos multidimensionnel à base des fourmis. Des expérimentations ont été faites, et ont donné des résultats satisfaisants en fonction des différents paramètres de l'algorithme de colonie des fourmis.

Expression de l'heuristique ou visibilité pour explorer l'espace de recherche par les fourmis : La métaheuristique basée sur les fourmis est une méthode probabiliste qui utilise simultanément les deux éléments fondamentaux pour passer d'une probabilité de transition à une autre. Ces deux éléments sont respectivement la phéromone τ déposée par les fourmis qui s'émergent en exploitant l'apprentissage au sein de la colonie, et l'heuristique η qui sert à explorer l'espace de recherche. La probabilité de transition est formalisée par l'équation 1. [1]

La phéromone τ , prend ses valeurs dans le cadre standard de l'algorithme. Elle se met à jour localement par le phénomène d'évaporation connu, et globalement se met à jour à l'aide des bonnes solutions obtenues itérativement par les fourmis. [1]

Pour l'exploration du champ de recherche, nous avons définis une heuristique qui est égale au rapport d'efficacité dont le dénominateur est égal au poids et le numérateur est égal à l'utilité, comme définie ci-dessus.

¹ P = P.Class_Level='A1DGFSPJTJ473'

² Le modèle ÖZSU = conjonction de plusieurs prédicats qui assure la complétude du schéma de fragmentation.

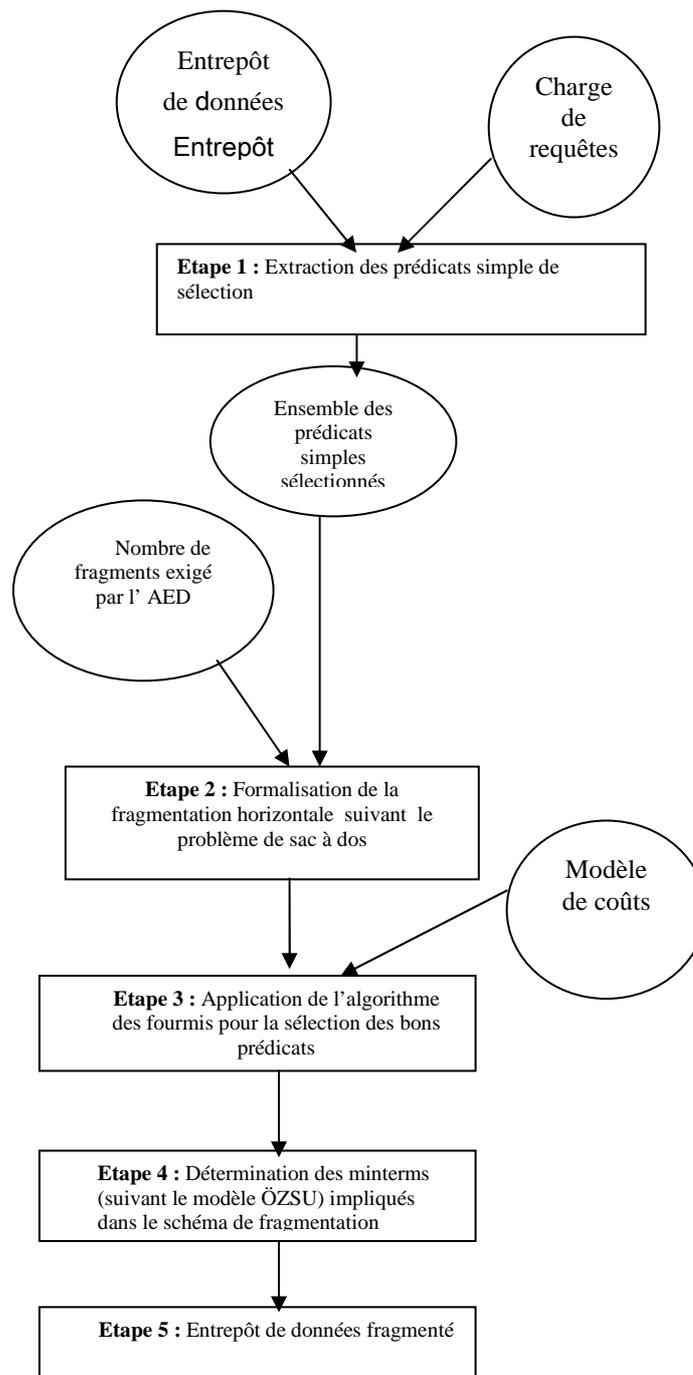


Figure 2 Architecture générale de notre solution

Et pour simplifier la valeur de l'heuristique η , nous développons le rapport d'efficacité (RE) comme suit :

Soit $A = (|F| * L) / PS$

posons R (Ressource) = $Sél(P_i) * A$

Utilité $U = A - R$

Le rapport d'efficacité (RE) relatif aux objets (prédicats) à remplir dans le sac.

$RE = U/R$

$$RE = A - Sél(P_i) * A / Sél(P_i) * A$$

$$RE = A(1 - Sél(P_i)) / A * Sél(P_i)$$

$$RE = (1 - Sél(P_i)) / Sél(P_i)$$

$$RE = 1 / Sél(P_i) - 1$$

RE = nombre d'enregistrements (F) / nombre d'enregistrements (P_i) - 1

$$RE = |F| / |F_{ri}| - 1$$

$$RE = 1 / Sél(P_i) - 1$$

$$RE = 1 - Sél(P_i) / Sél(P_i)$$

$$\begin{aligned} RE &= Sél(Non Pi) / Sél(Pi) \\ \eta &= RE \end{aligned}$$

N.B : $||F||$ = cardinalité de F, $||Fri||$ = taille d'un fragment introduisant un prédicat Pi, L=Longueur d'un enregistrement, PS=Taille d'une page disque, Non Pi = forme négative du prédicat Pi.

Algorithme de sac à dos par les fourmis appliqué au problème de fragmentation horizontale :

// En entrée : Entrepôt non fragmenté, Ensemble de prédicats(Sélectivité) ou de fragments
Initialiser les traces de phéromone à τ max

Répéter
Pour chaque fourmi k dans 1..NbAnts, construire une solution S_k comme suit :
Choisir aléatoirement un premier objet³ $o_1 \in 1..n$
 $S_k \leftarrow \{o_1\}$
Candidats $\leftarrow \{o_i \in 1..n / o_i \text{ peut être sélectionné sans violer des contraintes de ressources}\}$

Tant que Candidats $\neq \emptyset$ OR nombre de fragments non atteint
Faire
Choisir un objet $o_i \in$ Candidats avec la probabilité
$$P_{S_k}^4(o_i) = \frac{\tau |S_k(o_i)|^\alpha * |\eta S_k(o_i)|^\beta}{\sum_{o_i \in \text{Candidats}} \tau |S_k(o_i)|^\alpha * |\eta S_k(o_i)|^\beta}$$

$S_k \leftarrow S_k \cup \{o_i\}$
Enlever de Candidats chaque objet qui viole contraintes de ressources

Fin Faire
Fin Pour
Mettre à jour les traces de phéromone en fonction de $\{S_1, \dots, S_{nbAnts}\}$
Si une trace de phéromone est inférieure à τ min alors la mettre à τ min
Si une trace de phéromone est supérieure à τ max alors la mettre à τ max
Jusqu'à un nombre maximal de cycles atteint ou solution optimale trouvée

Paramétrage de l'algorithme :

Les deux paramètres essentiels qui caractérisent l'algorithme de colonie de fourmis sont :

- la phéromone déposée qui sert à guider les fourmis aux choix des objets à mettre dans le sac à l'aide de la concentration de ces derniers en phéromone ;

³ Un objet dans notre cas représente un prédicat simple de sélection
⁴ $P_{S_k}(o_i)$ représente la probabilité de sélectionner l'objet O_i par la fourmi K dans la solution S. Avec τ la phéromone, α =l'importance de la phéromone, et β =l'importance de la visibilité η . NbAnts = Nombre total de fourmis

- la visibilité ou heuristique η qui sert à explorer l'espace de recherche.

Le premier paramètre est généralement initialisé à la même valeur pour l'ensemble des objets afin de leur donner la même chance pour être sélectionné. Pour la visibilité représentée par le rapport d'efficacité dans l'équation 1, elle montre que les prédicats ayant des petites sélectivités seront les mieux choisis car ils sont reliés aux partitions qui présentent le moins de poids.

4.6. Phénomène de stagnation

Le principe de l'algorithme de fourmis est basé sur l'apprentissage transmis au sein de la colonie des fourmis à l'aide de la phéromone déposée, en outre les fourmis utilisent la visibilité pour choisir d'autres objets pertinents. La négligence de la visibilité entraîne un phénomène connu qui s'appelle le piégeage dans les optimums locaux. Les expérimentations que nous avons fait sur le benchmark APB1, ont montré que la négligence de la visibilité a conduit à une dégradation de la solution d'un taux d'amélioration par rapport à la charge globale des requêtes à 14%. La figure 3 suivante illustre ce phénomène.

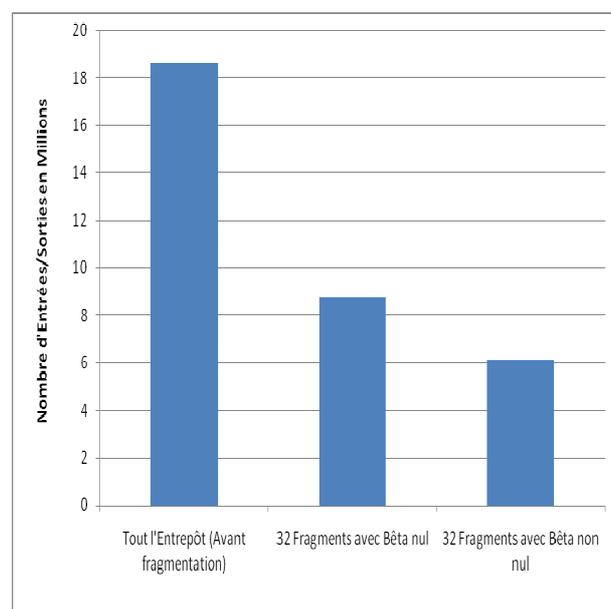


Figure 3 Dégradation de la solution pour la négligence de visibilité (facteur de l'heuristique Bêta nul)

Pour mieux comprendre ce phénomène de piégeage via la figure précédente, nous rappelons l'équation de la probabilité de la transition suivante :

$$P_{Sk}^5(o_i) = \frac{\tau |Sk(o_i)|^{\alpha} * | \eta Sk(o_i) |^{\beta}}{\sum_{o_i \in \text{Candidats}} \tau |Sk(o_i)|^{\alpha} * | \eta Sk(o_i) |^{\beta}}$$

Équation 1 Formule de la probabilité de transition

D'après la formule de la probabilité de transition, la négligence de la visibilité η , revient à fixer le facteur d'importance de visibilité β à zéro.

La figure montre bien que le nombre d'entrées/sorties est plus important pour β égal à zéro.

5. Benchmark utilisé

Pour mener à bien, notre étude nous avons utilisé le benchmark APB1, dont le modèle logique de données suivant :

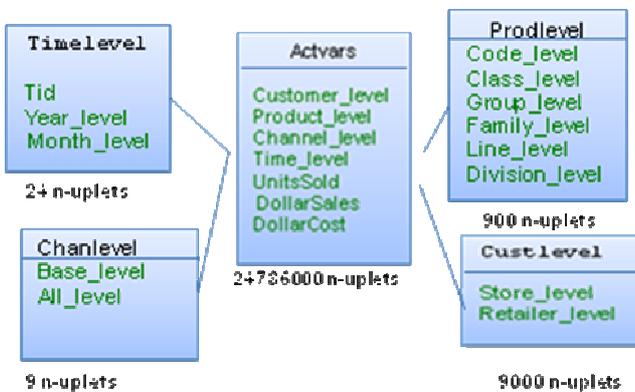


Figure 4 : Modèle logique de données du benchmark APB1

5.1. Exemple de poids des prédicats simples tirés du benchmark

Pour se conformer aux données du benchmark APB1, nous avons implémenté ce dernier sous le SGBD Oracle 10g, et nous avons calculé les sélectivités de tous les prédicats simples qui figurent sur la charge des requêtes constituée de 55 requêtes. Le tableau suivant illustre les sélectivités de cinq (05) prédicats, ainsi que le nombre d'entrées/sorties entre disque et mémoire centrale des requêtes introduisant ces prédicats.

5.2. Résultats d'expérimentation

Nous avons déroulé l'algorithme des fourmis pour déterminer le schéma de fragmentation le plus optimal possible, en estimant à chaque fois le taux de réduction par rapport au coût global de la charge des 55 requêtes utilisée.

Les premiers résultats obtenus sont illustrés dans le tableau suivant :

Ancien Nombre de fragments	Taux de réduction du coût global / Charge de requêtes
32	46,87 %
64	50,37 %
128	56,97 %

Tableau 1. Taux d'amélioration en fonction du nombre de fragments

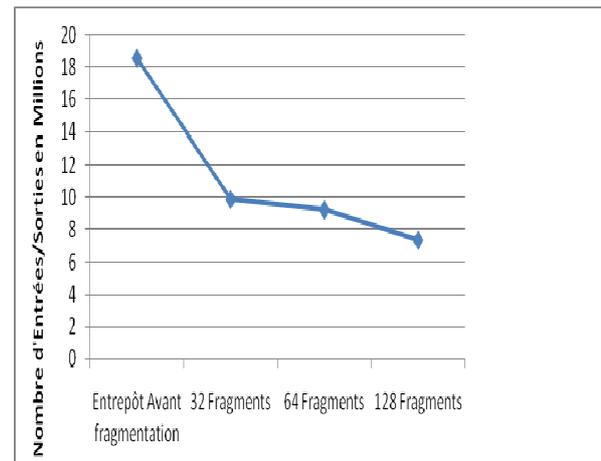


Figure 5 Représentation graphique des résultats du tableau précédent

5.3. Problèmes rencontrés

Parmi les problèmes que nous avons rencontrés, nous citons la volumétrie de certaines partitions après la fragmentation de la table des faits.

Nous rappelons dans ce contexte que les travaux qui ont été réalisés dans la prise en charge de la technique de fragmentation horizontale ne traitent pas le problème des partitions qui sont volumineuses, par conséquent les requêtes qui attaquent ces partitions mettent toujours un temps considérable pour acquérir les informations voulues.

A travers notre travail, nous avons formalisé une méthode basée sur une analyse de la charge des requêtes et qui consiste à éclater les partitions volumineuses en plusieurs sous partitions par rapport à l'attribut qui participe plus dans la charge des requêtes.

Dans notre cas, nous avons remarqué que l'attribut « Month_level » de la table de dimension temps est un bon axe d'analyse pour appliquer notre méthode d'éclatement.

En effet, l'éclatement de la partition volumineuse générée suite au schéma de fragmentation obtenu a participé à une amélioration totale de 14% par rapport à la charge globale en entrées/sorties des toutes requêtes.

Le tableau suivant affiche les nouveaux résultats obtenus.

5 $P_{Sk}(o_i)$ représente la probabilité de sélectionner l'objet O_i par la fourmi K dans la solution S . Avec τ la phéromone, α =l'importance de la phéromone, et β =l'importance de la visibilité η . NbAnts = Nombre total de fourmis

Ancien Nombre de fragments	Nombre de fragments après éclatement	Taux de réduction du coût global / Charge de requêtes
32	40	62,70 %
64	72	64,94 %
128	133	63,23 %

Tableau 2 Taux d'amélioration suite à la répartition du fragment volumineux

5.4. Validation pratique du modèle utilisé

La fonction objectif de notre travail est la minimisation du nombre d'entrées/sorties entre disque et mémoire centrale durant l'exécution des requêtes décisionnelles de la charge, en se basant sur la technique de fragmentation horizontale.

Les résultats des expérimentations obtenus ont montré l'efficacité de notre méthode.

Et pour valider pratiquement les résultats obtenus, nous avons fragmenté l'entrepôt du benchmark sous oracle10g. Puis nous avons calculé le temps de réponse pour l'ensemble des 55 requêtes décisionnelles.

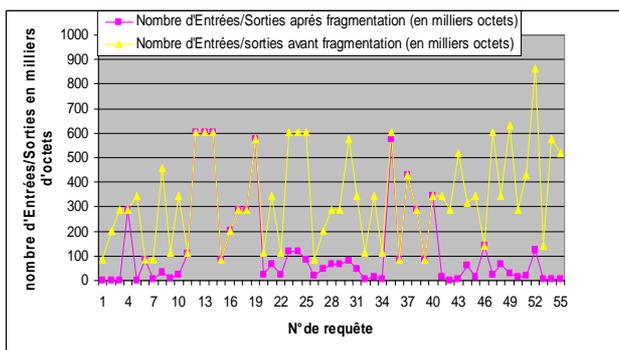


Figure 6 Courbe représentative du nombre d'E/S par requête de la charge

Le calcul sur machine du temps de réponse de l'ensemble des requêtes pour le cas de quarante (40) fragments, a abouti à un résultat de 60,61 % de gain global, et ceci pour le même schéma de fragmentation horizontale de quarante (40) fragments dont la fonction objectif est la minimisation des entrées/sorties entre mémoire et disque lors de l'exécution des requêtes décisionnelles.

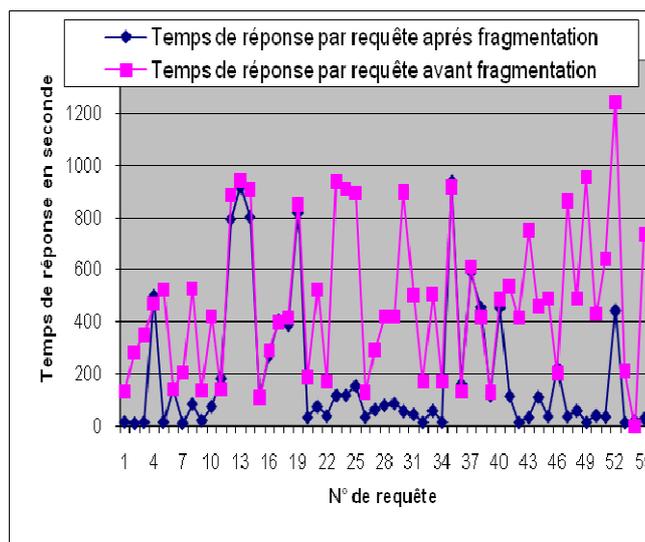


Figure 7 Courbe représentative du temps de réponse par requête de la charge

L'équivalence des résultats en nombre d'entrées/sorties (gain de 62,7 %, tableau 4 – figure 6), et en temps de réponse (60,61%, figure 6) confirme ce qui suit :

- le modèle d'estimation de coût utilisé est précis.
- L'écart léger de 2,09% (62,7% – 60,61%), est justifié par les coûts de jointures des requêtes que nous avons négligé dans le modèle d'estimation.

Conclusion

A la fin de ce travail, nous pouvons tirer les conclusions suivantes :

- Le cadrage du problème de sélection de la technique d'optimisation de la fragmentation horizontale, selon le problème de sac à dos, nous semble un champ de recherche très fructueux.
- La prise en charge du problème de fragmentation horizontale en se basant sur l'approche des fourmis, est une opportunité intéressante qui permet la solution du problème en un temps fini.
- La possibilité de généralisation de notre approche pour couvrir les structures des index et des vues matérialisées, est un sujet d'une grande importance.

Bibliographie

[1] E. Bonabeau, M. Dorigo, G. Theraulaz, "From Natural to Artificial Systems", Oxford University Press US, 1999 – 307 pages
 [2] H. Mahboubi, «Optimisation de la performance des entrepôts de données XML par fragmentation et répartition », thèse de doctorat, Université Lumière Lyon 2, 08/12/2008

- [3] K. Aouiche, « Sélection automatique d'index dans les entrepôts de données », rapport de recherche, Laboratoire ERIC Université Lumière Lyon 2 École Doctorale de Sciences Cognitives, le 08 décembre 2005.
- [4] K. Sol « Le Problème du Sac à Dos », Université de Montpellier 2, Juin 2007
- [5] L. Bellatreche, « Utilisation des Vues Matérialisées, des Index et de la Fragmentation dans la Conception Logique et Physique d'un Entrepôt de Données », Thèse de doctorat, université Clermont-Ferrand II Décembre 2000.
- [6] L. Bellatreche et K. Boukhalfa « An Evolutionary Approach to Schema Partitioning Selection in a Data Warehouse Environment », 7th International Conference on Data Warehousing and Knowledge Discovery (DAWAK'05)(3589), edited by Lecture Notes in Computer Science (LNCS), August, Springer-Verlag, 2005.
- [7] I. Alaya, C. Solnon, K. Ghedira, « Algorithme des fourmis avec différentes stratégies phéromonales pour le sac à dos multidimensionnel », rapport de recherche, Institut Supérieur de Gestion de Tunis, 2005