Pour qu’un hypergraphe soit un hyperarbre, il faut prouver que son hypergraphe dual est lui-même alpha-acyclique. Afin de prouver cela il faut montrer que le graphe primal de notre hypergraphe est cordal et que toute clique maximale est une hyperarête dans l’hypergraphe.  
Il nous faut donc tout d’abord des fonctions créant l’hypergraphe dual ainsi que le graphe primal \*explication ?\*

Afin de prouver que le graphe primal est cordal nous utilisons le principe d’ordonnancement d’élimination parfaite, en effet un graphe est cordal s’il possède un tel ordonnancement.

Pour cela, selon Rose, Lueker & Tarjan (1976), la cordialité d’un graphe peut être vérifiée efficacement grâce au principe de recherche lexicographique en largeur (LexBFS). [2]

Il faut donc tout d’abord réaliser une recherche lexicographique en largeur qui est un algorithme produisant une liste ordonnée de l’ensemble des sommets du graphe.  
Il fonctionne ainsi : A chaque itération le premier sommet de l’ensemble des sommets est choisi. Ensuite pour chaque sommet voisin avec ce sommet choisi et est placé dans un nouvel ensemble précédant le premier. L’algorithme poursuit ainsi jusqu’à ce que l’entièreté des sommets soient traités.  
Ensuite, il faut vérifier que cet ordonnancement est d’élimination parfaite. Pour cela il faut que dans la liste ordonnée, pour tous les sommets précédant un sommet « v » dans cette même liste, ils doivent former une clique. Sinon le graphe n’est pas cordal.

La deuxième partie qu’il faut prouver est que pour chaque clique maximale du graphe corresponde un hyperarête dans l’hypergraphe. L’algorithme de Bron-Kerbosch [3] permet de résoudre ce problème.  
Basiquement cet algorithme va chercher grâce à un backtracking récursif, l’ensemble des cliques maximales du graphe. A chaque fois qu’une clique est trouvée, nous vérifions si effectivement elle correspond à l’une des hyperarêtes de l’hypergraphe (l’hypergraphe dual en occurrence nous donne directement les informations dont nous avons besoin). Si ce n’est pas le cas l’algorithme s’arrête et renvoie un résultat erroné, sinon il continue jusqu’à la fin de l’exécution de l’algorithme.

Si ces deux parties sont vérifiées, alors notre hypergraphe est bien un hyperarbre.

[1] https://en.wikipedia.org/wiki/Chordal\_graph

[2] <https://en.wikipedia.org/wiki/Lexicographic_breadth-first_search>

[3] https://en.wikipedia.org/wiki/Bron%E2%80%93Kerbosch\_algorithm