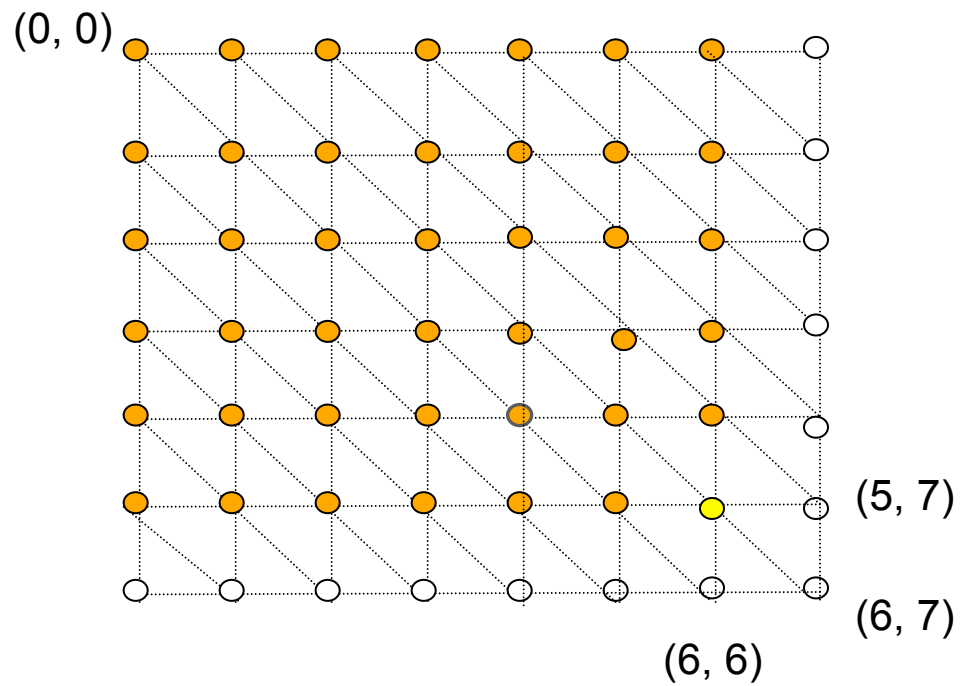
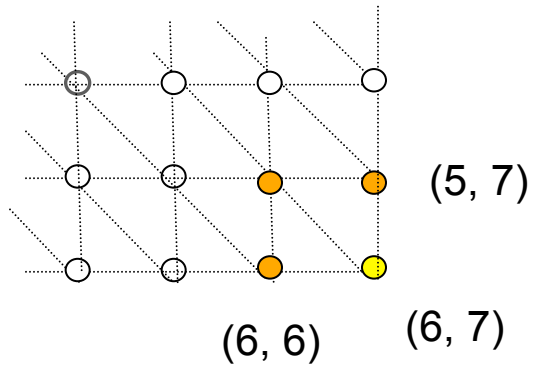
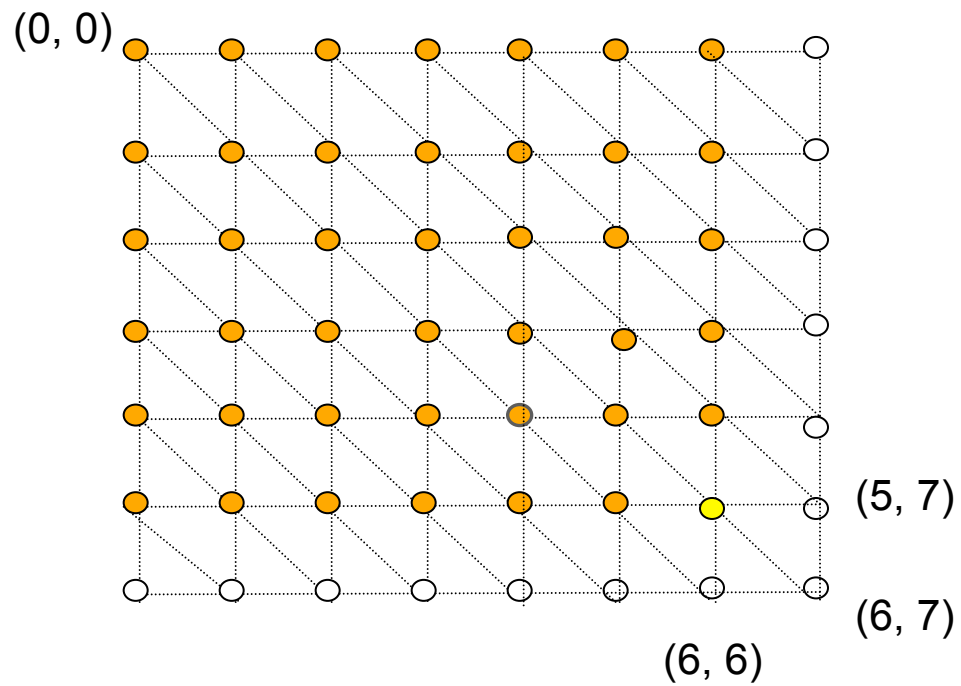
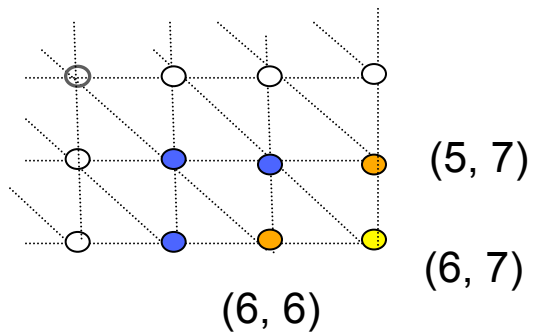
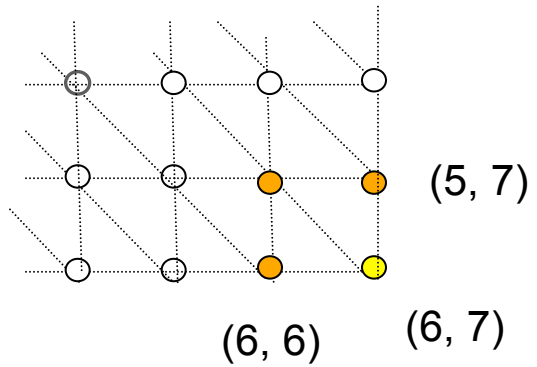


4. Comparaison de séquences

- Comment prédire les fonctions des gènes/protéines ?
- Évolution et similarité de séquences
- Quantifier la similarité de deux séquences
- L'alignement de séquences devient un problème d'optimisation
- Un alignement de séquences vu comme un chemin dans une grille
- Si un chemin est optimal, tous ses chemins partiels sont optimaux
- Coûts et alignement
- Un algorithme récursif
- **Eviter la récursivité : une version itérative**
- Cet algorithme est-il efficace ?

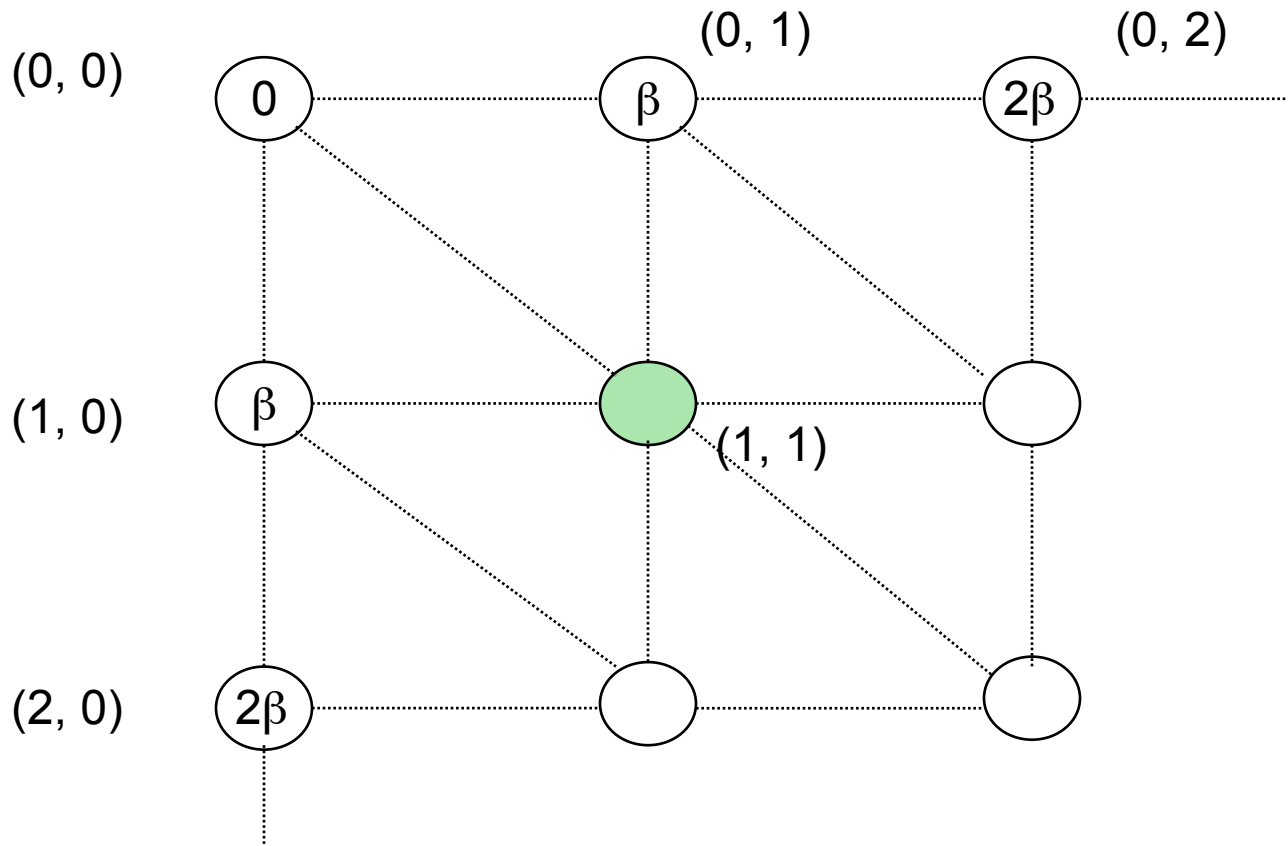


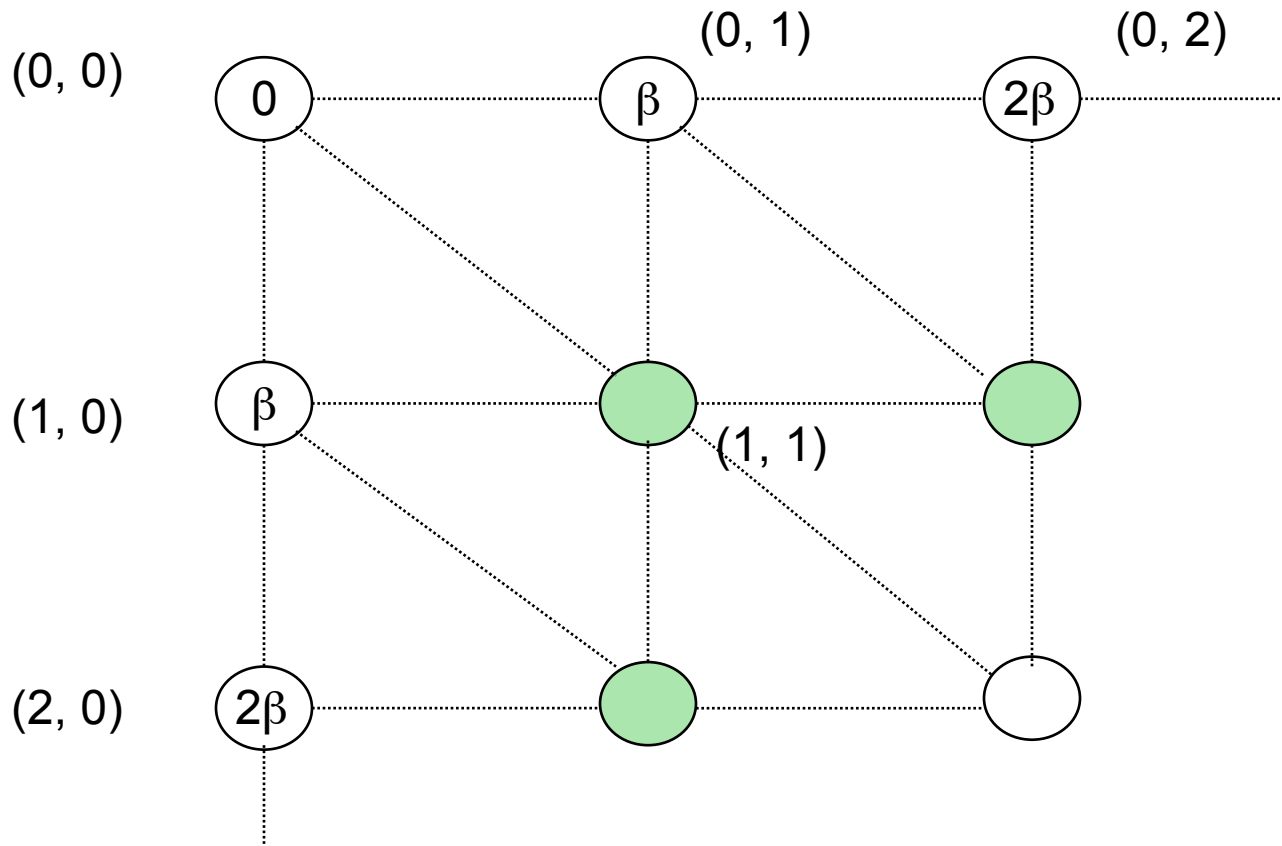


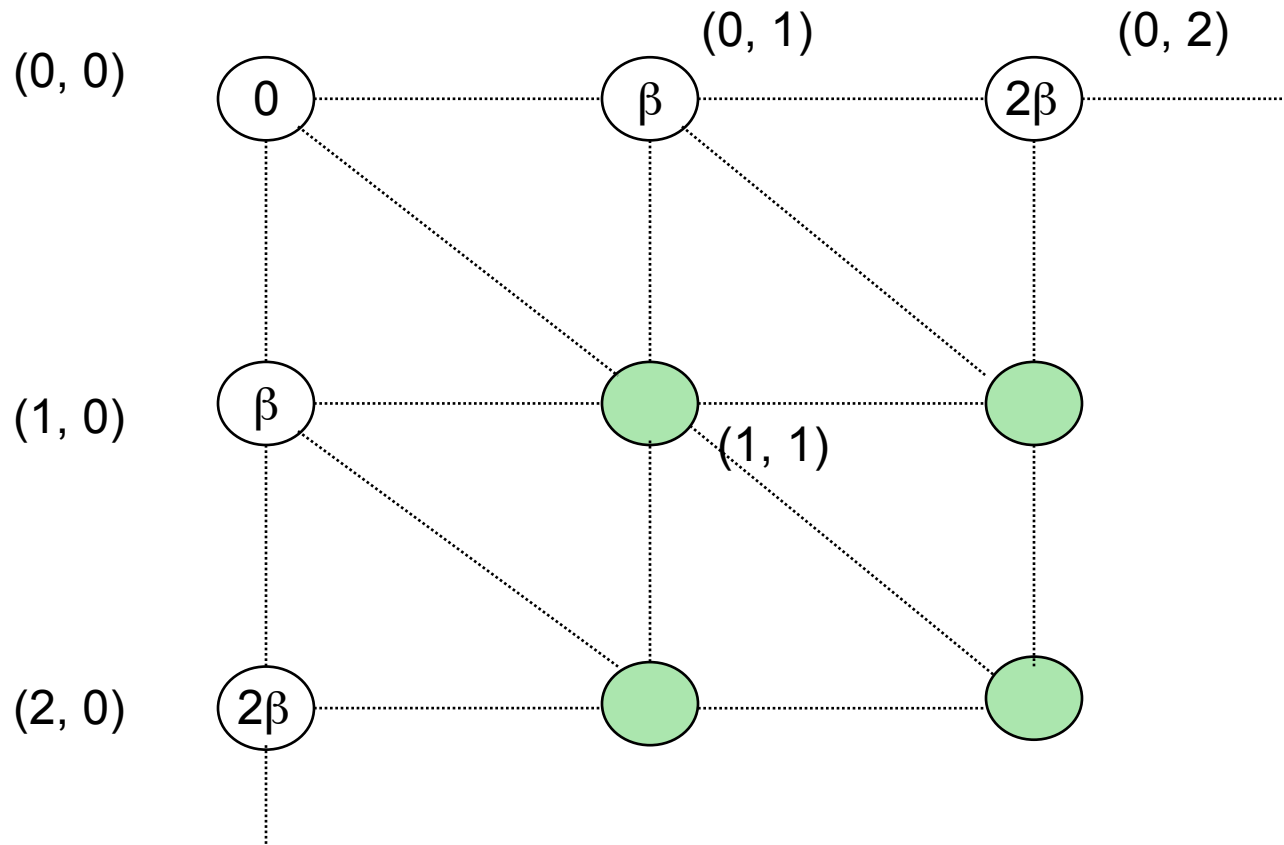
Un algorithme itératif en deux phases

- **Première phase**

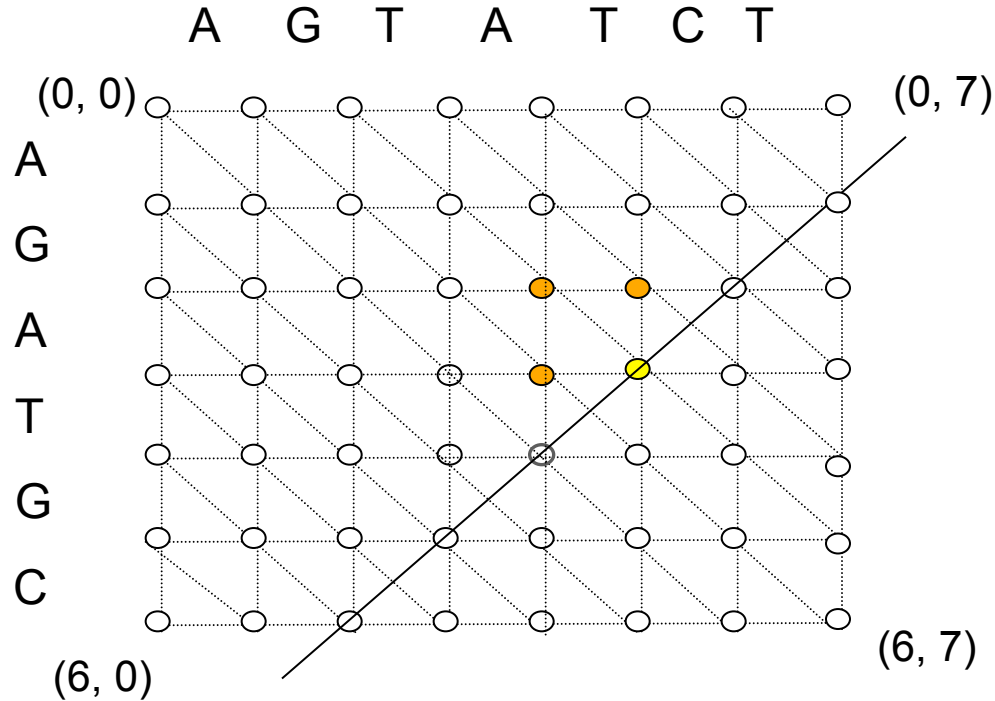
- Calcul du coût du chemin optimal qui aboutit sur chaque nœud $[I, J]$, en partant du nœud $[0, 0]$ jusqu'au nœud $[N, M]$
- Enregistrer ces coûts dans une matrice de dimensions $[0:N, 0:M]$







Calcul en parallèle possible des coûts d'une même diagonale



Un algorithme itératif en deux phases

- **Première phase**

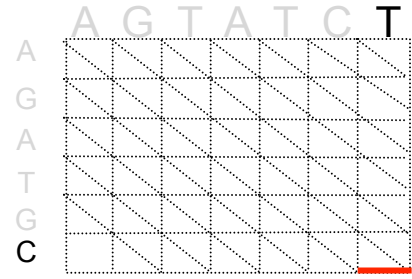
- Calcul du coût du chemin optimal qui aboutit sur chaque nœud $[I, J]$, en partant du nœud $[0, 0]$ jusqu'au nœud $[N, M]$
- Enregistrer ces coûts dans une matrice de dimensions $[0:N, 0:M]$

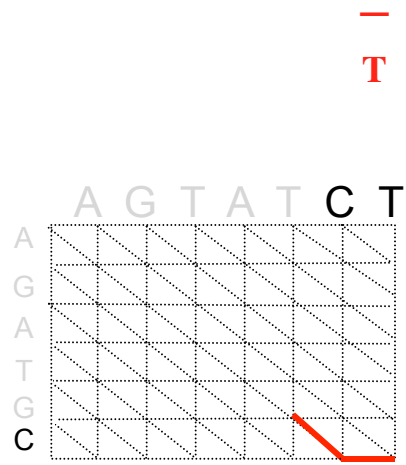
- **Seconde phase**

- Détermination du chemin optimal, en partant du nœud $[N, M]$ et en « remontant » jusqu'au nœud $[0,0]$, en utilisant les coûts optimaux calculés lors de la première phase

—
T

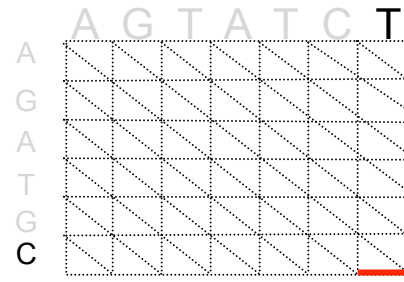
(a)





—
T

(a)



C —
C **T**

(b)

