# Produits minimaux de transpositions

Marc Lorenzi

13 février 2021

**Résumé**

Il est bien connu que toute permutation est un produit de transpositions. La parité du nombre de transpositions est un invariant, mais pas le nombre de transpositions intervenant dans le produit. Quel est le nombre minimal de transpositions nécessaires pour engendrer une permutation donnée ? On se propose de le calculer. On en déduit des bornes inférieures et supérieures de ce nombre de transpositions pour engendrer des permutations appartenant à certains sous-ensembles de $\mathfrak{S}_n$. On étudie enfin un algorithme de tri effectuant un nombre optimal d'échanges de variables. Afin que cette lecture soit profitable à tous, on termine par quelques exercices laissés au lecteur.

## 1 Décompositions minimales en produit de transpositions

### 1.1 Introduction

Dans tout ce qui suit $n$ désigne un entier naturel supérieur ou égal à 2. Pour toute permutation $\sigma \in \mathfrak{S}_n$, on note $m(\sigma)$ le nombre d'orbites selon $\sigma$. Pour tout cycle $\gamma$ on note $\ell(\gamma)$ la longueur du cycle.

**Remarque.** Par exemple, $m(id) = n$. Si $\gamma$ est un $p$-cycle, alors $m(\gamma) = n - p + 1 = n - \ell(\gamma) + 1$. Pour une permutation $\sigma$ quelconque écrite $\sigma = \gamma_1 \dots \gamma_q$ comme un produit de cycles de supports disjoints, on a $m(\sigma) = q + n - \sum_{i=1}^{q} \ell(\gamma_i)$.

### 1.2 Le lemme clé

**Lemme 1.** Soit $\sigma \in \mathfrak{S}_n$. Soient $i, j \in [\![1, n]\!]$ deux entiers distincts. Soit $\tau = (ij) \in \mathfrak{S}_n$ une transposition. Soit $\sigma' = \sigma\tau$. On a $m(\sigma') = m(\sigma) \pm 1$.

**Démonstration.** Soit $k \in [\![1, n]\!]$ différent de $i$ et de $j$. On a alors $\sigma(k) = \sigma'(k)$. Les orbites selon $\sigma$ ne contenant ni $i$ ni $j$ sont donc les mêmes que les orbites selon $\sigma'$ ne contenant ni $i$ ni $j$. Considérons maintenant ce qui se passe pour les orbites contenant $i$ ou $j$. Il y a deux cas à considérer selon que $i$ et $j$ sont ou pas dans la même orbite selon $\sigma$. Dans les figures ci-dessous, les orbites selon $\sigma$ sont représentées en traits pleins et celles selon $\sigma'$ sont représentées en pointillés.

- Cas 1 : $i$ et $j$ sont dans la même orbite $\mathcal{O}$ selon $\sigma$.
  Dans ce cas, l'orbite $\mathcal{O}$ se transforme en deux orbites selon $\sigma'$. On a $m(\sigma') = m(\sigma) + 1$.
- Cas 2 : $i$ et $j$ ne sont pas dans la même orbite selon $\sigma$
  Ces deux orbites se transforment en une unique orbite selon $\sigma'$. On a $m(\sigma') = m(\sigma) - 1$.

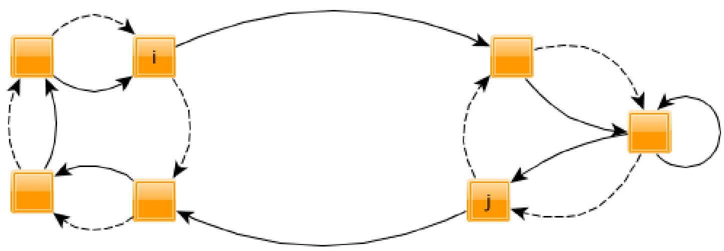


FIGURE 1 – $i$ et $j$ dans la même orbite selon $\sigma$

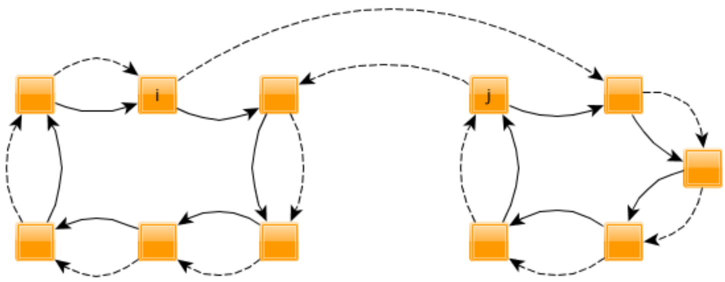


FIGURE 2 – $i$ et $j$ dans deux orbites différentes selon $\sigma$

| Cas | $m(\sigma')$ | $n - m(\sigma')$ |
|---|---|---|
| 1 | $m(\sigma) + 1$ | $n - m(\sigma) - 1$ |
| 2 | $m(\sigma) - 1$ | $n - m(\sigma) + 1$ |

□

### 1.3 Conséquences

**Corollaire 2.** Soit $\sigma \in \mathfrak{S}_n$. Soit $\tau \in \mathfrak{S}_n$ une transposition. Soit $\sigma' = \sigma\tau$. Alors $n - m(\sigma') \geq n - m(\sigma) - 1$.

**Démonstration.** Lire la dernière colonne du tableau ci-dessus. □

**Proposition 3.** Soit $\sigma \in \mathfrak{S}_n$. On suppose que $\sigma$ est le produit de $k$ transpositions, où $k \in \mathbb{N}$. Alors, $k \geq n - m(\sigma)$

**Démonstration.** On fait une récurrence sur $k$. Pour $k = 0$, on a $\sigma = id$. Et $n - m(\sigma) = 0$. On a donc bien $k \geq n - m(\sigma)$. Supposons maintenant la propriété vérifiée pour un entier $k$. Soit $\sigma = \tau_1 \ldots \tau_{k+1}$ un produit de $k+1$ transpositions. Soit $\sigma' = \sigma\tau_{k+1}$. On a $\sigma' = \tau_1 \ldots \tau_k$. Par l'hypothèse de récurrence, $k \geq n - m(\sigma')$. Le lemme précédent nous dit que cette quantité est supérieure ou égale à $n - m(\sigma) - 1$. Ainsi, $k + 1 \geq n - m(\sigma)$ et la propriété est vérifiée pour l'entier $k + 1$. □

**Proposition 4.** Soit $\sigma \in \mathfrak{S}_n$. Il existe une décomposition de $\sigma$ en produit de $n - m(\sigma)$ transpositions.

**Démonstration.** Écrivons $\sigma = \gamma_1 \ldots \gamma_p$, où les $\gamma_i$ sont des cycles de supports disjoints. La permutation $\sigma$ possède $p$ orbites non triviales et $n - \sum_{k=1}^{p} \ell(\gamma_k)$ orbites réduites à un point.

Ainsi, $m(\sigma) = n - \sum_{k=1}^{p} \ell(\gamma_k) + p$ et $n - m(\sigma) = \sum_{k=1}^{p} \ell(\gamma_k) - p$. Remarquons maintenant que tout cycle $\gamma = (x_1 \dots x_\ell)$ s'écrit facilement comme un produit de $\ell - 1$ transpositions, par exemple $\gamma = (x_1x_2)(x_2x_3)\dots(x_{\ell-1}x_\ell)$. En appliquant ceci à chacun des cycles $\gamma_i$ on obtient une décomposition de $\sigma$ en produit de transpositions, qui sont au nombre de $\sum_{i=1}^{p}(\ell(\gamma_i) - 1) = \sum_{i=1}^{p} \ell(\gamma_i) - p = n - m(\sigma)$. □

Pour $\sigma \in \mathfrak{S}_n$, on note $E(\sigma)$ l'ensemble des entiers $k$ tels que $\sigma$ est le produit de $k$ transpositions. Notons $\lambda(\sigma) = \min E(\sigma)$. La quantité $\lambda(\sigma)$ est le nombre minimal de transpositions dont $\sigma$ est le produit. Les deux propriétés précédentes montrent :

**Proposition 5.** Pour toute permutation $\sigma \in \mathfrak{S}_n$, on a $\lambda(\sigma) = n - m(\sigma)$.

Remarquons pour terminer le résultat suivant :

**Proposition 6.** $E(\sigma) = \{\lambda(\sigma) + 2k, k \in \mathbb{N}\}$.

**Démonstration.** Les résultats sur la signature d'une permutation nous disent que la parité du nombre de transpositions intervenant dans la décomposition d'une permutation en produit de transpositions est un invariant. On a donc $E(\sigma) \subseteq \{\lambda(\sigma) + 2k, k \in \mathbb{N}\}$. Inversement, soit $k$ un entier naturel. Soit $\tau$ une transposition quelconque. Soit $\sigma = \tau_1 \dots \tau_{\lambda(\sigma)}$ une décomposition minimale de $\sigma$ en produit de transpositions. Alors $\sigma = \tau_1 \dots \tau_{\lambda(\sigma)}\tau^{2k}$ est une décomposition de $\sigma$ en produit de $\lambda(\sigma) + 2k$ transpositions. Donc $\lambda(\sigma) + 2k \in E(\sigma)$. □

### 1.4 Permutations à support de cardinal fixé

Soit $\sigma \in \mathfrak{S}_n$. On appelle support de $\sigma$ l'ensemble $\{i \in [\![1, n]\!], \sigma(i) \neq i\}$. On note $q(\sigma)$ le cardinal du support de $\sigma$. On note $\mathfrak{S}_n^q$ l'ensemble des permutations de $\mathfrak{S}_n$ dont le support est de cardinal $q$. En d'autres termes, $\mathfrak{S}_n^q = \{\sigma \in \mathfrak{S}_n, q(\sigma) = q\}$. Par exemple, $\mathfrak{S}_n^0 = \{id\}$, $\mathfrak{S}_n^1 = \emptyset$, pour tout $q > n$, $\mathfrak{S}_n^q = \emptyset$.

**Exercice 1.** $\mathfrak{S}_n^2$ est l'ensemble des transpositions. $\mathfrak{S}_n^3$ est l'ensemble des 3-cycles. À partir de $q = 4$, les choses se compliquent ...

**Proposition 7.** Soit $q \in [\![2, n]\!]$. On a

$$\min\{\lambda(\sigma), \sigma \in \mathfrak{S}_n^q\} = \lfloor \frac{q+1}{2} \rfloor$$

et

$$\max\{\lambda(\sigma), \sigma \in \mathfrak{S}_n^q\} = q - 1$$

**Démonstration.** Soit $\sigma \in \mathfrak{S}_n^q$. La permutation $\sigma$ possède $n - q$ points fixes, donc $n - q$ orbites réduites à un point. Comme $q \geq 2$, $\sigma$ a au moins un point non fixe (et même deux), donc au moins orbite non triviale. Donc $m(\sigma) \geq n - q + 1$ et ainsi $\lambda(\sigma) = n - m(\sigma) \leq n - (n - q + 1) = q - 1$. Prenons maintenant pour $\sigma$ un $q$-cycle. On a alors $m(\sigma) = n - q + 1$, donc $\lambda(\sigma) = q - 1$.

Passons maintenant à l'examen du minimum. Appelons $p(\sigma)$ le nombre d'orbites selon $\sigma$ non triviales (i.e. non réduites à un point). Une orbite non triviale pour $\sigma$ contient au moins deux points et les orbites sont disjointes, il y a donc au moins $2p(\sigma)$ points dans de telles orbites. Mais ces points étant non fixes, il y en a au plus $q$. On a donc $2p(\sigma) \leq q$ d'où $p(\sigma) \leq \frac{q}{2}$ et

$m(\sigma) \leq n - q + \frac{q}{2} = n - \frac{q}{2}$. Ainsi, $\lambda(\sigma) = n - m(\sigma) \geq \frac{q}{2}$. Comme $\lambda(\sigma)$ et $q$ sont entiers on en déduit facilement que $\lambda(\sigma) \geq \lfloor \frac{q+1}{2} \rfloor$. L'entier $\lfloor \frac{q+1}{2} \rfloor$ minore donc l'ensemble qui nous intéresse. Pour finir la démonstration distinguons selon la parité de $q$.

- Supposons que $q = 2q'$ est pair. Soit $\sigma$ un produit de $q'$ transpositions de supports disjoints. On a alors $m(\sigma) = q' + n - q = n - q'$ d'où $\lambda(\sigma) = q'$. Cela tombe bien puisque $q' = \lfloor \frac{q+1}{2} \rfloor$.
- Supposons que $q = 2q' + 1$ est impair. Soit $\sigma$ un produit de $q' - 1$ transpositions de supports disjoints et d'un 3-cycle dont le support est disjoint de celui des transpositions. On a $\sigma \in \mathfrak{S}_n^q$ et $m(\sigma) = (q' - 1) + 1 + n - q = q' + n - q = n - q' - 1$ d'où $\lambda(\sigma) = q' + 1$. Or $q' + 1 = \lfloor \frac{q+1}{2} \rfloor$.

□

# 2 Un algorithme de tri optimal pour les échanges de variables

## 2.1 Introduction

Soit $(A, \leq)$ un ensemble totalement ordonné. Nous nous intéressons dans cette section au problème du tri dans l'ordre croissant d'un tableau d'éléments de $A$, avec comme règle que les modifications apportées aux éléments du tableau ne peuvent être faites qu'en échangeant deux éléments dans le tableau. La plupart des algorithmes de tri classiques fonctionnent sur ce principe. Pour n'en citer que quelques-uns, les tris par bulle, par insertion, par sélection, le tri rapide, le tri par fusion, le tri par tas. Certains algorithmes de tri comme le tri par baquets ou le tri radix fonctionnent sur d'autres principes, comme par exemple des méthodes de comptage et l'affectation directe d'une valeur à une case du tableau.

## 2.2 Notations

Pour simplifier les raisonnements nous ne considérerons que des tableaux dont tous les éléments sont distincts. Pour un tel tableau $t$, il existe une unique permutation des éléments du tableau qui trie le tableau. Nous noterons cette permutation $\sigma^t$. Nous appellerons également degré de mélange du tableau l'entier $\lambda(t) = \lambda(\sigma^t)$. C'est le nombre minimal d'échanges de deux éléments de $t$ nécessaire pour trier le tableau.

Dans toute la suite, nous noterons de façon non classique $\mathfrak{S}_n$ l'ensemble des permutations de $[\![0, n-1]\!]$. Ceci nous permet de faire agir plus aisément les permutations sur les tableaux qui, dans beaucoup de langages de programmation, ont leurs indices numérotés à partir de 0.

Soit $\sigma \in \mathfrak{S}_n$. Soit $t = [x_0, x_1, \ldots, x_{n-1}]$ un tableau de taille $n$. On note $t \otimes \sigma$ le tableau $t \otimes \sigma = [x_{\sigma(0)}, \ldots, x_{\sigma(n-1)}]$. Dit autrement, pour tout $k \in [\![0, n-1]\!]$, $(t \otimes \sigma)[k] = t[\sigma(k)]$. Par exemple, le tableau obtenu en échangeant les éléments d'indices $i$ et $j$ du tableau $t$ est $t \otimes (ij)$. Cette notation bizarre a l'avantage de fournir une propriété d'associativité :

**Proposition 8.** Soit $t$ un tableau de taille $n$. Soient $\sigma, \sigma' \in \mathfrak{S}_n$. On a alors

$$t \otimes (\sigma\sigma') = (t \otimes \sigma) \otimes \sigma'$$

**Démonstration.** Notons $t' = t \otimes \sigma$ et $t'' = t' \otimes \sigma'$. On a alors pour tout $k \in [\![0, n-1]\!]$ :

$$\begin{aligned}((t \otimes \sigma) \otimes \sigma')[k] &= t''[k] \\ &= (t' \otimes \sigma')[k] \\ &= t'[\sigma'(k)] \\ &= (t \otimes \sigma)[\sigma'(k)] \\ &= t[\sigma(\sigma'(k))] \\ &= t[(\sigma\sigma')(k))] \\ &= (t \otimes (\sigma\sigma'))[k]\end{aligned}$$

Ceci étant vrai pour tout $k$, les deux tableaux $(t \otimes \sigma) \otimes \sigma'$ et $t \otimes (\sigma\sigma')$ sont égaux.

□

## 2.3 Quelques remarques

Avec ces nouvelles notations supposons $\sigma^t = \tau_1 \dots \tau_k$ décomposée en produit de transpositions. Le tri du tableau $t$ est donc réalisé en calculant $t_1 = \tau_1.t$, puis $t_2 = \tau_2.t_1$, ..., puis $t_k = \tau_k.t_{k-1}$. Le tableau $t_k$ est alors le tableau $t$ trié. Le seul souci, c'est que pour un tableau $t$ donné on ne connaît pas $\sigma^t$ ...

Revenons un instant sur le tableau à la fin de la démonstration du lemme de la section I. On y remarque que pour diminuer la quantité $\lambda(\sigma)$ il faut et il suffit que l'on multiplie $\sigma$ par une transposition $(ij)$ telle que $i$ et $j$ appartiennent à une même orbite selon $\sigma$. Si l'on réussit à trouver un tel $i$ et un tel $j$, alors $t' = t \otimes (ij)$ a la propriété intéressante que $\lambda(t') = \lambda(t) - 1$. D'où le :

**Proverbe du jour** : *pour trier le tableau $t$ efficacement il faut exploser les orbites de $\sigma^t$.*

Un algorithme réussissant à trouver un tel $i$ et un tel $j$ sur un tableau $t$ diminue $\lambda(t)$ de 1 à chaque échange d'éléments. Ainsi, après $\lambda(t)$ échanges, l'algorithme fournit un tableau dont le $\lambda$ est nul, bref un tableau trié. Et qui plus est un tableau trié après un nombre d'échanges d'éléments minimal.

Il s'avère qu'un tel algorithme est bien connu. Il s'agit du tri par sélection.

## 2.4 Le tri par sélection

Soit à trier un tableau $t$ de taille $n$. Pour $i$ variant de 0 à $n-1$ on cherche le plus petit élément de $t$ entre les indices $i$ et $n-1$. Soit $j$ l'indice de cet élément : si $j \neq i$, on échange les éléments de $t$ d'indices $i$ et $j$. Sinon, on va prendre un café.

On peut montrer par récurrence sur $i$ qu'à la fin de la $i$ème itération, les éléments du tableau $t$ d'indices 0 à $i$ sont les $i+1$ plus petits éléments de $t$ dans l'ordre croissant. S'il y a effectivement échange à la fin de la $i$ème itération, cela signifie donc que la « bonne » position pour $t[j]$ est en fait la position $i$. Autrement dit, $\sigma^t(j) = i$, et donc $i$ et $j$ sont dans une même orbite selon $\sigma^t$. On a gagné. Le tri par sélection trie tout tableau avec un nombre d'échanges d'éléments minimal.

```
def tri(t):
  n = len(t)
  for i in range(n):
    j = i
```

```
for k in range(i, n):
  if t[k] < t[j]: j = k
if i != j: t[i], t[j] = t[j], t[i]
```

**Exercice 2.** Calculer, en fonction de $n$, le nombre de comparaisons d'éléments de tableau effectuées par la fonction `tri` sur un tableau de taille $n$.

## 2.5 Exercices

1. Déterminer $\lambda(\sigma)$ pour toutes les permutations $\sigma$ de $\mathfrak{S}_4$ et $\mathfrak{S}_5$.
2. Déterminer un tableau $t$ tel que $\sigma^t = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 9 & 6 & 2 & 4 & 1 & 7 & 3 & 8 & 5 & 0 \end{pmatrix}$. Déterminer $\lambda(t)$.
3. On considère le tableau $t = [1, 4, 2, 8, 5, 7]$.
   (a) Calculer $\sigma^t$. En déduire $\lambda(t)$.
   (b) Simuler l'exécution de `tri(t)` en complétant le tableau ci-dessous. La ligne étiquetée $i$ contient les informations sur ce qui se passe lors de la $i$ème itération de la fonction `tri` ($i = 0 \dots 5$). Dans la colonne étiquetée $t$, indiquer le contenu du tableau $t$ *après* l'itération numéro $i$. Dans la colonne étiquetée $j$, indiquer s'il y a lieu l'indice $j$ pour lequel il y a échange de $t[i]$ et $t[j]$ lors de l'itération $i$.

| $i$ | $j$ | $t$ |
|---|---|---|
| 0 | – | $[1, 4, 2, 8, 5, 7]$ |
| 1 | 2 | $[1, 2, 4, 8, 5, 7]$ |
| 2 | | |
| 3 | | |
| 4 | | |
| 5 | | $[1, 2, 4, 5, 7, 8]$ |

4. Déterminer un tableau $t$ tel que $\sigma^t \in \mathfrak{S}_{12}^9$ et $\lambda(t) = 8$.
5. Déterminer un tableau $t$ tel que $\sigma^t \in \mathfrak{S}_{12}^9$ et $\lambda(t) = 5$.
6. Modifier la fonction de tri pour qu'elle renvoie le nombre d'échanges effectués lors du tri par sélection. Tester cette fonction sur les tableaux des deux exercices précédents.