

Espace coloré de Topsxi

Petite introduction au cours

Toute la force de l'espace coloré de Topsxi en tant que telle réside dans le lien étroit qu'il fait naître entre les différents domaines. L'informatique joue un rôle majeur dépassant la simple reprise du code hexadécimal. Un rôle qui ouvre la voie vers une autre vision des couleurs.

Partie 3 : Opérateurs logiques

IV Chapitre IV : Opérateurs logiques

Dans l'espace coloré de Topsxi, on définit trois types d'opérations :

- les **opérations arithmétiques**, à savoir l'addition et la soustraction (*la multiplication sera introduite bien plus tard*)
- les **opérations géométriques**, comme les homothéties, les symétries, les rotations, etc
- les **opérations logiques** qui viennent tout droit de l'algèbre booléenne

Dans ce chapitre, nous allons définir pleinement les 7 opérateurs logiques.

IV. I L'opérateur *NOT*

IV. IA Définition algébrique

L'opérateur *NOT* renvoie la couleur complémentaire d'une couleur c . On dit que $\tilde{c}$ est le complémentaire de c si la somme de leur composante deux à deux vaut exactement FF_{16} . Les notations sont les suivantes :

$$NOT(c) = \tilde{c} \text{ ou } \sim c = \tilde{c}$$

Autrement dit, *NOT* vérifie la propriété suivante : $\forall c \in \mathfrak{C}_T, c + (\sim c) = W$.

Pour calculer le complémentaire d'une couleur, on associe le **complémentaire chiffre hexadécimal par chiffre** du code couleur original. On peut s'aider du tableau de correspondance ci-dessous.

Fig. Les couples NOT

$$| 0 \leftrightarrow F | 1 \leftrightarrow E | 2 \leftrightarrow D | 3 \leftrightarrow C | 4 \leftrightarrow B | 5 \leftrightarrow A | 6 \leftrightarrow 9 | 7 \leftrightarrow 8 |$$

Comment lire le tableau ?

$\sim 5 = ?$: on regarde le tableau, on a $5 \leftrightarrow A$ donc $\sim 5 = A$

Par exemple :

$$\begin{aligned} \#2351C9 &\xrightarrow{NOT} \#DCAE36 \\ NOT(\#851230) &= \#7AEDCF \\ \sim \#A307F5 &= \#5CF80A \end{aligned}$$

IV. IB Définition géométrique

=> : Théorème du NOT

Effectuer un *NOT* revient à effectuer une symétrie centrale de centre le centre de l'espace coloré de Topsxi transmuté.

=>: Démonstration

Soit c une couleur associée au point coloré $A(x, y, z)$.

Le complémentaire $\tilde{c}$ de c est par définition de couleur complémentaire associé au point $\tilde{A}$ de coordonnées $(FF - x, FF - y, FF - z)$.

Cherchons le milieu du segment $[A\tilde{A}]$ que l'on nommera I .

Par le théorème du milieu, I a pour coordonnées :

$$\left(\frac{x + (FF - x)}{2}, \frac{y + (FF - y)}{2}, \frac{z + (FF - z)}{2} \right)$$

soit $\left(\frac{FF}{2}, \frac{FF}{2}, \frac{FF}{2} \right)$.

On remarque tout de suite que I n'appartient pas à l'espace coloré de Topsis.

Transmutons alors $\mathfrak{E}_T$. Dans le monde réel, I a pour coordonnées $\left(\frac{255}{2}, \frac{255}{2}, \frac{255}{2} \right)$, soit $(127.5, 127.5, 127.5)$. I est donc le centre de symétrie transformant A en $\tilde{A}$.

Or, I est aussi le centre du cube $[0, 255]^3$ contenant l'espace coloré de Topsis transmuté.

Ce raisonnement s'applique pour tout point coloré A . D'où la proposition.

CQFD

La démonstration précédente soulève un point important :

=>: Théorème de complémentarité

Toute couleur possède un complémentaire distinct.

=>: Démonstration

D'après le théorème du NOT, toute couleur possède un complémentaire différent de lui-même par symétrie (puisque I est le centre de $\mathfrak{E}_T$ transmuté).

Seul I est son propre complémentaire. Or $I \notin \mathfrak{E}_T$ donc I n'est pas une couleur.

CQFD

IV. IC Propriétés héritées du NOT

- $\forall c \in \mathfrak{E}_T, \sim(\sim c) = c$

IV. II L'opérateur AND

IV. IIA Définitions préliminaires

Définition: On appelle **filtre élémentaire** les composantes suivantes (en base 16) :

01 02 04 08 10 20 40 80

Nous pouvons remarquer que les filtres élémentaires possèdent **un chiffre nul et un chiffre de la forme 2^i** .

Les filtres élémentaires jouent un rôle important dans la définition du AND.

IV. IIB Définition algébrique

L'opérateur AND est l'équivalent d'un **filtre soustractif** d'une couleur d appliqué à une couleur c .

On note $AND(c, d) = c'$ ou $c \& d = c'$ avec c' la couleur c filtré soustractivement par d .

=>: Théorème algorithmique du AND

- Calculer $AND(c, d)$ revient à appliquer un AND composante par composante.
- Si une composante d_j est un filtre élémentaire, alors le chiffre 2^i devient
 - 0 si le chiffre correspondant de c a son modulo 2^{i+1} strictement inférieur à 2^i
 - 2^i sinon

L'autre chiffre est ramené à 0.

- Si la composante d_j n'est pas un filtre élémentaire, alors on décompose d_j en filtres élémentaires.

Exemple : calculons $163593 \& 07C6F5$:

1. On regarde composante par composante (1er cas du théorème algorithmique du AND)

$$16\&07 \quad 35\&C6 \quad 93\&F5$$

2. On divise les composantes de d en filtres élémentaires (3e cas du théorème algorithmique du AND)

$$16\&(04 + 02 + 01) \quad 35\&(80 + 40 + 04 + 02) \quad 93\&(80 + 40 + 20 + 10 + 04 + 01)$$

3. On distribue et on applique le 2e cas du théorème algorithmique du AND :

$$\begin{aligned} 16\&04 & \begin{cases} 1\&0 \rightarrow 0 \\ 6[4 \times 2] \geq 4? \xrightarrow{\text{oui}} 4 \end{cases} \rightarrow 04 \\ 16\&02 & \begin{cases} 1\&0 \rightarrow 0 \\ 6[2 \times 2] \geq 2? \xrightarrow{\text{oui}} 2 \end{cases} \rightarrow 02 \\ 16\&01 & \begin{cases} 1\&0 \rightarrow 0 \\ 6[1 \times 2] \geq 1? \xrightarrow{\text{non}} 0 \end{cases} \rightarrow 00 \end{aligned}$$

d'où $16\&07 = 04 + 02 + 00 = 06$.

De la même façon,

$$\begin{array}{rcl} & 93\&80 = 80 & \\ & 93\&40 = 00 & \\ 35\&80 = 00 & 93\&20 = 00 & \\ 35\&40 = 00 & 93\&10 = 10 & \\ 35\&04 = 04 & 93\&04 = 00 & \\ 35\&02 = 00 & 93\&01 = 01 & \\ \hline 35\&C6 = 04 & 93\&F5 = 91 & \end{array} \quad (1)$$

4. On regroupe les composantes : $163593 \& 07C6F5 = 060491$

Cet algorithme peut paraître long, mais combiné intelligemment avec les propriétés d'après, il est possible d'aller plus vite.

=> : Démonstration (non optimale) du théorème algorithmique du AND

Soit c et d deux couleurs.

1. Comme le AND est un opérateur bit par bit, nous pouvons tout bêtement calculer $AND(c, d)$ bit par bit (bit = chiffre en base 2). Mais nous pouvons faire le choix de regrouper les bits entre eux, de sorte à faire le calcul composante par composante.
2. Supposons $d = 2^i 00000$. Alors tous les chiffres de $c\&d$ valent 0 (0 est l'élément absorbant du AND) sauf lorsque le chiffre vaut 2^i .
Dans ce cas, on peut démontrer la proposition par exhaustivité des cas. Ou bien en remarquant que pour un chiffre c_i de c inférieur à 2^i , on a 0 et pour c_i supérieur à 2^i (éventuellement avec le modulo à la puissance supérieure), on obtient la même valeur.
3. AND est une opération bit par bit. En réorganisant les bits, on peut faire sortir deux nombres avec des bits distincts d_1 et d_2 tels que $d_1 + d_2 = d$.
On peut également remarquer qu'un filtre élémentaire s'écrit avec un seul chiffre 1 en base 2 (sous réserve d'écrire en Binary Coded Hexadecimal le filtre).
Or, l'addition est distributive sur le AND , si les nombres distribués ont des bits distincts deux à deux, ce que respectent les filtres élémentaires.
En répétant ainsi le processus jusqu'à ce que les nombres d_k soient tous des filtres élémentaires, on prouve le 3e cas.

CQFD

IV. IIC Propriétés du AND

Soit b et d deux chiffres hexadécimaux.

Propriétés héritées :

- $b \& b = b$
- $b \& F = b$
- $b \& 0 = 0$

Propriétés supplémentaires :

- $b \& d = d \& b$
- si $d = d_1 + d_2$, alors $b \& d = b \& d_1 + b \& d_2$

Ces propriétés permettent d'aller plus vite.

Par exemple, $93 \& F5$ peut s'écrire tout simplement $93 \& (F0 + 04 + 01)$, sachant que $93 \& F0 = 90$ d'après les propriétés juste au-dessus. C'est nettement plus rapide que de décomposer $F0$ en $80 + 40 + 20 + 10$.

IV. IID Définition géométrique

=>: Théorème géométrique du AND

Soit $c' = AND(c, d)$ avec $c, d \in \mathfrak{E}_T$.

c' est la position qu'occuperait c si l'espace coloré de Topsxi était réduit à un cube de sommet opposé à l'origine d .

Autrement dit, l'opération $AND(c, d)$ réduit $\mathfrak{E}_T$ en un cube de diagonale $\sqrt{3} \times d(K, d)$ et renvoie les coordonnées de c dans ce nouveau cube.

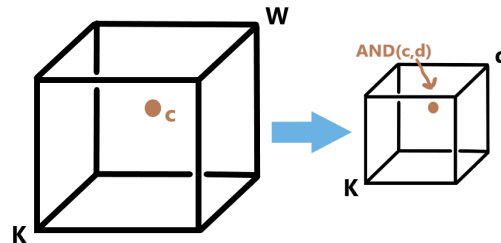


Figure 1: Illustration de l'opération $AND(c, d)$

=>: Démonstration

La propriété du modulo indique non seulement une division de l'espace mais également un caractère répétitif. Ainsi, nous pouvons diviser $\mathfrak{E}_T$ en plusieurs petits cubes identiques. Cette unité est caractérisée par une zone où tous les points sont de couleur noire. Cette zone est localisée sur l'origine. Le reste du cube est un ensemble de sommes de combinaisons de filtres élémentaires.

Par modulo, ce petit cube est répété sur le reste de l'espace. Nous pouvons donc considérer uniquement le cube dont un sommet est l'origine K .

Pour connaître la couleur associée au point opposé à l'origine, nous calculons :

$$AND(W, d) = AND(FFFFFFF, d) = d.$$

Ainsi, $AND(c, d)$ donne les coordonnées de c dans ce cube unité.

À noter que le cube peut être déformé.

CQFD

IV. III Les opérateurs XOR et $XNOR$

Pour simplifier, nous introduirons ces deux opérateurs logiques ensemble (et avant l'opérateur OR).

IV. IIIA Définition algébrique du XOR

L'opérateur XOR est l'équivalent d'un **filtre additif** de couleur d appliqué à une couleur c .
On note $XOR(c, d) = c \oplus d = c'$ où c' est la couleur c filtrée additivement par d .

Contrairement à l'opérateur AND , une formule a été trouvée pour l'opérateur XOR .

=>: Formule de Xorros

Soit b et d deux **chiffres hexadécimaux**.

Si b et d sont tous les deux **pairs**, alors :

$$b \oplus d = (b + 2[d \pmod{4} \geq 2?] \times (-1)^{[b \pmod{4} \geq 1?]} + 4[d \pmod{8} \geq 4?] \times (-1)^{[b \pmod{8} \geq 4?]} + 8[d \geq 8?]) \pmod{16}$$

La réponse aux questions entre crochets $[]$ est binaire : 1 si c'est vrai, 0 si c'est faux.

=>: Démonstration

Se référer au repo sur GitHub (démonstration par exhaustivité des cas) :

> fichier xorros_demo_exhaustive.py

Exemple d'application directe :

$$\begin{aligned} 4 \oplus 6 &= (4 + 2[6 \pmod{4} \geq 2?] \times (-1)^{[4 \pmod{4} \geq 1?]} + 4[6 \pmod{8} \geq 4?] \times (-1)^{[4 \pmod{8} \geq 4?]} + 8[6 \geq 8?]) \pmod{16} \\ &= (4 + 2 \times 1 \times (-1)^0 + 4 \times 1 \times (-1)^1 + 8 \times 0) \pmod{16} \\ &= (4 + 2 - 4 + 0) \pmod{16} \\ &= 2 \pmod{16} = 2_{16} \end{aligned}$$

Efficace bien que robuste !

À noter que la formule de Xorros ne fonctionne **que pour deux chiffres pairs** !

=>: Théorème algorithmique du XOR

Soit l'opération $XOR(c, d)$ entre deux points colorés c et d .

- Effectuer $XOR(c, d)$ revient à effectuer un XOR chiffre hexadécimal par chiffre entre c et d .
- Si les deux chiffres sont pairs, nous utiliserons la formule de Xorros.
- Si les **deux** chiffres sont **impairs**, alors $c_i \oplus d_i = (c_i - 1) \oplus (d_i - 1)$.
- Si les deux chiffres sont **de parité différente**, alors $c_i \oplus d_i = c_i \oplus (d_i - 1) + 1$

=>: Démonstration :

1. Cette démonstration se base sur le même principe que celle du premier point du théorème algorithmique du AND , à savoir le fait qu'il s'agit avant tout d'un opérateur bit à bit.
2. se référer à la démonstration de la formule de Xorros
3. Soit c_i et d_i deux chiffres impairs des couleurs c et d .
Nous passons en base 2 pour la démonstration :
 c_i peut s'écrire $u_3u_2u_11$ et d_i $v_3v_2v_11$. Le dernier chiffre est forcément un 1 puisque c_i et d_i sont tous les deux impairs. Les autres chiffres peuvent être des 0 ou des 1.
Or, $c_i - 1$ s'écrit forcément $u_3u_2u_10$. De même pour d_i $v_3v_2v_10$. Dans ce cas, si nous appliquons le XOR bit par bit, le résultat pour les trois premiers bits ne change pas puisque les bits ont été conservés.

En revanche, le dernier bit des deux chiffres a changé. Cela dit, nous pouvons remarquer que $0 \oplus 0 = 1 \oplus 1 = 0$.

Ainsi, le dernier bit du résultat ne change pas.

Étant donné que les quatre bits du résultat de $XOR(c_i, d_i)$ sont conservés, alors la formule $c_i \oplus d_i = (c_i - 1) \oplus (d_i - 1)$ est vérifiée pour tous chiffres hexadécimaux impairs.

4. Soit c_j et m_j deux chiffres hexadécimaux **pairs**. Alors avec la formule de Xorros, nous obtenons $x_j = XOR(c_j, m_j)$.

Comme c_j et m_j sont tous les deux pairs, alors le dernier bit de x_j vaut forcément $0 \oplus 0 = 0$.

Si nous considérons le chiffre $m_j + 1$, celui-ci diffère de m_j que par son dernier bit qui vaut 1 au lieu de 0. Puisque c_j est toujours pair (donc son dernier bit vaut 0), alors le dernier bit de x_j devient $0 \oplus 1 = 1$.

Ainsi, $c_j \oplus (m_j + 1) = c_j \oplus m_j + 1$. En posant $d_j = m_j + 1$ (d_j impair donc), nous arrivons à $c_j \oplus d_j = c_j \oplus (d_j - 1) + 1$

CQFD

IV. IIIB Définition algébrique du $XNOR$

En informatique, on définit l'opérateur $XNOR$ comme $NOT(XOR)$. Cette définition ne change pas pour l'espace coloré de Tpsxi.

Ainsi, calculer $XNOR(c, d) = c \odot d$ revient à calculer $NOT(XOR(c, d)) = \sim (c \oplus d)$. Nous pouvons alors nous reporter sur toutes les formules et propriétés de l'opérateur XOR .

IV. IIIC Propriétés des opérateurs XOR et $XNOR$

Propriétés héritées : Soient c et d deux couleurs.

- $c \oplus 000000 = c \oplus K = c$
- $c \oplus FFFFFFFF = c \oplus W = \sim c$
- $c \oplus c = 000000 = K$
- $c \oplus (\sim c) = FFFFFFFF = W$
- $(\sim c) \oplus (\sim d) = c \oplus d$
- $c \oplus d = d \oplus c$
- $(c \oplus d) \oplus d = c$

Nous en déduisons les propriétés de $XNOR$ par utilisation du NOT .

Propriétés combinées :

- $(c \odot d) \oplus d = \sim c$
- $(c \oplus d) \odot d = \sim c$

Une propriété supplémentaire importante pour la suite :

$$c \oplus d = \sim (\sim (c \oplus d)) = \sim (c \odot d)$$

IV. IIID Définitions géométriques des opérateurs XOR et XNOR**=> : Théorème géométrique du XOR**

Soit $c' = XOR(c, d)$.

Alors c' est la position qu'occuperait le symétrique de δ dans l'espace coloré de Topsxi réduit à un cube de diagonale (K, A)

où

δ est la position qu'occuperait le symétrique de c dans l'espace coloré de Topsxi réduit à un espace de première diagonale $(K, \tilde{d})$

et A vérifie $distance(K, A) = distance(\tilde{d}, \delta')$ où δ' est la position qu'occuperait c dans l'espace coloré de Topsxi réduit à un cube de diagonale (K, d) .

Remarque : Avec les nombreuses réductions, l'espace final n'est plus un cube. Toutefois, on conserve la notion de "première diagonale" comme "image de la diagonale principale du cube".

=> : Démonstration :

La démonstration repose sur l'égalité booléenne :

$$c \oplus d = \left(\sim \left((\sim c) \& (\sim d) \right) \right) \& \left(\sim (c \& d) \right)$$

(Les parenthèses sont surreprésentées pour bien discerner les priorités.)

Ensuite, on utilise les théorèmes géométriques des AND et NOT.

=> : Théorème géométrique du XNOR

Soit l'opération $XNOR(c, d) = c''$.

Alors c'' est le symétrique de $XOR(c, d)$ par rapport au centre de l'espace coloré de Topsxi transmuté.

IV. IIIE Notion avancée : la couleur de Clara

Intuitivement – et il est fort facile de le repérer – le résultat de $XOR(c, d)$ semble être un mixte entre une identité (ID) et une négation (NOT).

C'est ce qu'aborde le théorème suivant :

=> : Théorème de Clara Le Hen

Soit deux couleurs c et d .

Alors, $XOR(c, d) = ID(h) + |d - c|$ où h est appelée la **couleur de Clara de c et de d** .

La couleur de Clara représente la négation tandis que la valeur absolue représente l'identité.

La question est de savoir comment calculer cette couleur de Clara, qui notez le bien, dépend des **deux** valeurs de c et d .

Remarques :

- Clara Le Hen est une vraie personne qui a existé. Ce théorème est une mention (disons une dédicace) utilisée pour la simple raison créative. Cette femme n'a nullement (en tout cas au moment de l'écriture de ce PDF) participé à l'élaboration de ce théorème.
- En écrivant ce PDF, je me suis rendu compte que la construction de cette couleur est plus compliquée qu'il n'en paraît et qu'il faudrait sûrement définir d'autres choses. Pour cette raison, l'objet de la construction de la couleur de Clara fera l'objet d'un PDF bonus à part entière.

Le théorème de Clara Le Hen va plus loin :

=> : Corollaire du théorème de Clara Le Hen

Soit deux couleurs c et d .

Alors le nombre de F apparaissant dans d correspond au nombre de NOT à effectuer sur c .

Et le nombre de 0 apparaissant dans d correspond au nombre de ID à effectuer sur c .

=>: Ébauche de démonstration (démonstration détaillée dans l'annexe concernée) :

Soit c et d deux couleurs.

Écrits en base 2, nous pouvons calculer $c' = c \oplus d$ bit par bit.

Nous décomposons c' en h et m avec la règle suivante : si un bit de c' est le même que c , alors nous le comptons comme un bit de m . Si un bit de c' est différent (donc la contradiction (ou NOT) de celui de c), nous le comptons comme un bit de h . Nous complétons les "trous" par des 0.

Ainsi, nous obtenons h comme "négation partielle" de c et m comme "identité partielle" de c .

Il ne manquera plus qu'à démontrer que $m = |d - c|$.

Pour le corollaire, la démonstration repose sur les propriétés de l'opérateur XOR.

IV. IV L'opérateur OR

IV. IVA Définition algébrique

L'opérateur OR est l'équivalent d'un filtre deux tons de couleur d appliqué à une couleur c .

L'opération se note $OR(c, d) = c | d$.

Remarque : Ne pas confondre avec $|$ qui signifie "est divisible par" dans le monde réel.

Pour rappel, la division n'existe pas dans le monde naturel.

=>: Formule de Brookhaven : $\forall c, d \in \mathfrak{C}_T, c | d = c \& d + c \oplus d$

=>: Démonstration :

Nous passons en base 2.

Nous étudions tous les cas possibles :

		$\&$	$\oplus$
0	=	0	0
1	=	0	1
1	=	0	1
1	=	1	0

En groupant les bits par paquet de 4 pour les convertir en base 16, nous obtenons la formule de Brookhaven.

CQFD

=>: Conjecture de la simplification

Cette conjecture suppose que, si une formule explicite pour l'opérateur AND est trouvée, alors par simplification avec la formule de Xorros (opérateur XOR), la formule de Brookhaven donne une formule directe pour l'opérateur OR.

IV. IVB Propriétés de l'opérateur OR

Soit c une couleur.

Alors :

- $c | K = c | 000000 = c$
- $c | W = c | FFFFFFFF = W$
- $c | c = c$

Remarque : la formule de Brookhaven permet de retrouver ces propriétés à partir de celles des opérateurs AND et XOR.

IV. IVC Définition géométrique

Au moment de la rédaction de ce PDF, aucune définition géométrique n'a été donnée à l'opérateur OR. La raison est que la définition géométrique du XOR telle qu'établie plus haut n'est pas compatible à un phénomène de translation.

En effet, la formule de Brookhaven met en lumière un mouvement de translation à partir du point coloré $c \& d$ (ou inversement à partir du point coloré $c \oplus d$).

iv.v Les opérateurs *NAND* et *NOR*

Les opérateurs *NAND* et *NOR* sont les opérateurs réciproques des opérateurs logiques respectifs *AND* et *OR*.

Ils sont définis de la manière suivante :

- $NAND(c, d) = NOT(AND(c, d))$
- $NOR(c, d) = NOT(OR(c, d))$

Les propriétés et méthodes de calcul de ces opérateurs se déduisent de la combinaison de l'opérateur *NOT* avec l'opérateur adéquat.

Contrairement à l'opérateur *XNOR*, les opérateurs *NAND* et *NOR* ne jouent pas de rôle particulier, raison pour laquelle nous ne les détaillerons point dans ce PDF.