

Quand l'intelligence est juste une question de recopie

William Correa¹ Henri Prade¹ Gilles Richard¹

¹ IRIT, Université de Toulouse, France

{William.Correa, prade, richard}@irit.fr

Résumé

Cet article propose une approche générale qui permet de résoudre des tests de QI basés sur les matrices progressives de Raven. Dans ce type de test, on a généralement 8 cellules contenant des motifs géométriques, et il s'agit de sélectionner le contenu de la neuvième cellule parmi une série de 8 images candidates. L'approche proposée ici est basée sur une vue logique des proportions analogiques (c.-à-d., des énoncés de la forme 'A est à B ce que C est à D'). Cette approche peut être appliquée à une description basée sur des caractéristiques de haut niveau du contenu des cellules de la matrice de Raven, mais aussi dans certains cas à des niveaux de représentation très bas comme celui des pixels de l'image correspondante. L'approche repose sur un principe uniforme, qui calcule directement le contenu de la neuvième cellule (au lieu de sélectionner la solution parmi les choix proposés). Le principe revient pour l'essentiel à recopier des caractéristiques déjà observées (dans la mesure où on ne se heurte pas à des contradictions) ! Une discussion comparative avec d'autres travaux permet de souligner la généralité de l'approche.

Abstract

This paper proposes a general approach that is able to solve IQ tests based on Raven's progressive matrices, where the content of the 9th cell has to be selected in a set of candidate cells in order to complete the 8 other cells which already contain geometric features. The approach is based on a logical view of analogical proportions (i.e., statements of the form "A is to B as C is to D"). It applies to a feature-based description of the cells, and in a number of cases also applies at a very low level of representation i.e the pixel level ! Relying on a uniform principle, the approach allows us to compute the content of the 9th cell, rather than selecting it among proposed choices. It appears that this is just done by copying previously seen patterns ! A comparative discussion with other works is provided, emphasizing the generality of the approach.

1 Introduction

Depuis le début de l'intelligence artificielle, les tests de QI ont été vus comme des défis majeurs : ils impliquent en général des processus de raisonnement analogique. Ainsi, T. G. Evans [2, 3] fut le premier à concevoir un programme heuristique capable de sélectionner avec succès une réponse parmi plusieurs choix de manière à compléter des séquences de 3 images géométriques par une quatrième image. Le problème était alors de trouver, dans la collection des solutions potentielles, l'image X qui respectait le mieux la proportion analogique 'A est à B ce que C est à D'. Ce programme exploitait des représentations appropriées de figures géométriques indiquant les rapports entre sous-figures dans une figure donnée, et essayait de rapprocher les règles décrivant les changements pour aller de A à B avec celles décrivant les changements de C à X pour tous les X candidats. A l'époque, l'approche ne reposait pas sur une approche explicite formelle du raisonnement analogique, et était spécifiquement adaptée à la résolution de problèmes d'analogies géométriques.

D'une manière générale, deux types d'approches ont été proposées pour le raisonnement analogique : les unes symboliques, et les autres incorporant des caractéristiques connexionnistes. Les premières (e.g., [23][9]) utilisent des représentations symboliques appropriées pour les éléments impliqués, et essayent de les rapprocher afin de révéler des relations analogiques cachées. L'approche connexionniste [11][10][12] utilise des réseaux de satisfaction de contraintes, et détermine la meilleure solution parmi des hypothèses concurrentes au travers d'un processus d'optimisation. Ce dernier type d'approche incorpore une vue graduelle de la similarité, tandis que les approches symboliques ne dépassent pas la simple reconnaissance de l'identité. On pourra se reporter à [7] pour une discussion générale, dans laquelle l'auteur expose clairement que la capacité humaine pour observer qu'*'un objet ou une situa-*

tion particulière dans un contexte est semblable à un autre objet ou situation dans un autre contexte' est le processus principal à l'œuvre dans une analogie, et que cette capacité est une des caractéristiques principales de l'intelligence humaine.

Une vue logique des proportions analogiques, c.-à-d., des énoncés de la forme 'a est à b ce que c est à d', a été proposée récemment [15, 18]. Cette approche a permis de définir un processus d'inférence logique pour établir des analogies et en obtenir des conclusions plausibles. Dans ce processus, les entités sont décrites en termes de caractéristiques évaluées de manière binaire, et on recherche des relations analogiques parmi les entités, caractéristique par caractéristique. Le potentiel pour résoudre des tests de QI de cette vue logique des proportions analogiques a été suggéré sur différents types de problèmes [19], y compris dans le cas d'un test de Raven [21]. Les tests de Raven, qui sont largement utilisés dans la pratique en psychologie, ont la forme d'une série de problèmes tous avec le même format, c'est-à-dire une matrice de taille 3×3 en général dont les cellules contiennent des figures géométriques diverses, à l'exception de la dernière cellule qui est vide et doit être complétée en sélectionnant une solution parmi 8 figures candidates. Les séries de matrices utilisées dans un test sont plus ou moins progressives en difficulté (d'où la dénomination de 'Matrices progressives de Raven', abrégée en MPR dans la suite). Il est clair que l'application de proportions analogiques ou du raisonnement analogique dans ce cas est moins directe que dans le cas d'une succession de 3 figures qui doivent être complétées par une quatrième. Dans les deux dernières décennies, il y a eu quelques travaux [1] [13] qui ont cherché à résoudre automatiquement un grand nombre de tests de Raven. L'approche la plus ancienne [1] exploite des règles inspirées des pratiques humaines, qui décrivent les différents types de situations trouvées, alors que la plus récente [13] utilise une catégorisation plus abstraite des cas possibles de tests de Raven menant à un plus petit nombre de classes de situations potentielles.

Dans cet article, nous montrons qu'il est possible de résoudre presque tous les tests de Raven en utilisant une approche uniforme basée sur une vue logique des proportions analogiques, et applicables à différents niveaux de représentation pour certains problèmes. L'article est organisé de la manière suivante. Premièrement, un court résumé de la vue logique des proportions analogiques est fourni en mettant l'accent sur leur pouvoir prédictif. Ensuite l'approche proposée est présentée et illustrée sur différentes matrices de Raven. On insiste sur le fait qu'il y a des cas de problèmes de Raven qui peuvent être résolus avec une représentation de bas niveau en termes de pixels, tandis que d'autres ne peuvent être traités que par une représentation en termes de caractéristiques de haut niveau. Il est à remarquer que notre approche calcule le contenu de la dernière cellule des matrices Raven au lieu de le sélectionner parmi

plusieurs options. Il est aussi à noter que c'est en fait une vue réduite des proportions analogiques, présentée dans ce papier, qui est à l'œuvre. Cela revient dans les faits à recopier d'une manière contrôlée des changements (ou l'absence de changements) qu'on observe. Le papier se termine par une discussion comparative avec les travaux existants sur le sujet, soulignant en conclusion qu'au delà des tests de QI, les proportions analogiques ont aussi un intérêt pour la classification et l'apprentissage.

2 Proportions analogiques : Un résumé

Pour s'intéresser aux proportions analogiques, on peut commencer par considérer le cas des proportions numériques qui relient 4 valeurs a, b, c, d pour lesquelles on affirme une égalité de rapports sous la forme $\frac{a}{b} = \frac{c}{d}$, où le rapport $\frac{a}{b}$ est une représentation compacte des dissemblances entre a and b . Cette égalité nous dit que 'a diffère de b comme c diffère de d' et inversement. Quand on considère des valeurs booléennes au lieu de nombres, une manière simple d'abstraire ces idées est donnée en [15, 18] où des indicateurs de similarité et de dissemblance sont définis pour une paire booléenne (a, b) comme ci-dessous :

- $a \wedge b$ et $\bar{a} \wedge \bar{b}$ (indicateurs de *similarité*),
- $a \wedge \bar{b}$ et $\bar{a} \wedge b$ (indicateurs de *dissimilarité*).

2.1 Définition

D'une manière générale, une proportion logique est une paire d'équivalences entre des indicateurs. Parmi les 120 proportions différentes [16] qui existent, la proportion analogique "a est à b ce que c est à d" est définie comme :

$$a : b :: c : d = (a \wedge \bar{b} \equiv c \wedge \bar{d}) \wedge (\bar{a} \wedge b \equiv \bar{c} \wedge d)$$

Cette définition peut être vue comme la contrepartie logique de 'a diffère de b comme c diffère de d', et inversement. Ce qui est la signification attendue de "a est à b ce que c est à d". En tant que formule booléenne, la proportion analogique peut être comprise au travers de sa table de vérité que nous exhibons en Figure 1 (où les lignes menant à faux sont ignorées). Une caractéristique utile de cette

FIGURE 1 – Table de vérité de la proportion analogique

0	0	0	0
1	1	1	1
0	0	1	1
1	1	0	0
0	1	0	1
1	0	1	0

proportion, adéquate pour notre objectif principal qui est de construire un nouvel objet à partir des objets existants, est le fait que si on a seulement 3 éléments a, b, c , on peut trouver s'il existe un quatrième x unique pour compléter la proportion : c'est un problème de résolution d'équation,

fondamentalement similaire à la règle de 3 pour les proportions numériques. Selon les valeurs de a, b, c , l'équation n'est pas toujours résoluble. Un examen direct de la table de vérité donne la réponse, qui peut être formalisée comme suit pour les proportions analogiques : $a : b :: c : x$ est résoluble (en x) si et seulement si $((a \equiv b) \vee (a \equiv c)) = 1$. Dans ce cas, la seule solution x est $a \equiv (b \equiv c)$ [15]. En regard du problème de résolution d'équation, observons que la table de vérité ci-dessus peut être divisée en 2 parties :

- une partie contenant les lignes 0011 et 1100 : Dans ce cas, on comprend facilement que, lorsque l'on essaie, par exemple, de résoudre la proportion $0 : 0 :: 1 : x$, l'observation de la première paire (a, b) n'apporte aucun indice pour compléter la proportion parce que la valeur de c n'apparaît pas dans la paire (a, b) .

- une partie avec les 4 lignes restantes 0101, 1010, 1111 et 0000 : Il est clair que le troisième élément est égal au premier (i.e. $a \equiv c$), et le processus pour résoudre l'équation est réduit à la recopie du deuxième élément dans la quatrième position. La formule logique correspondant à la table de vérité avec seulement ces 4 lignes est

$$(a \equiv c) \wedge (b \equiv d)$$

Comme on pourra le voir dans la suite, il est remarquable que, pour résoudre les tests de Raven, ces 4 lignes constituent les patrons effectifs à prendre en compte. On est ainsi amené à penser que les tests de QI sont simplement une question de recopie !

2.2 Schémas analogiques généralisés

Quand il s'agit d'identifier des analogies à un haut niveau d'abstraction (c'est normalement le cas dans les tests de Raven), l'approche booléenne de base n'est pas suffisante. On peut cependant l'élargir de différentes manières.

Il est simple d'étendre la proportion analogique booléenne à des vecteurs booléens : on considérera qu'une proportion est valide pour des vecteurs booléens de dimension n , à savoir $\vec{a}, \vec{b}, \vec{c}, \vec{d}$, si et seulement si elle est valide composante par composante. Ce qui se formalise :

$$\vec{a} : \vec{b} :: \vec{c} : \vec{d} \text{ ssi } \forall i \in [1, n], a_i : b_i :: c_i : d_i$$

C'est une manière simple de considérer une proportion en établissant un rapport entre les objets décrits par de multiples caractéristiques de même nature. Mais on peut avoir besoin de plus. Revenons à l'univers numérique pour bien comprendre le problème. Supposons que nous devons compléter la séquence 2, 4, 19, ? d'une manière cohérente. Dans ce cas, nous sommes amenés à appliquer au troisième élément 19 la même fonction simple qui a été appliquée à 2 pour obtenir 4 (dans la mesure où cette fonction peut être identifiée sans ambiguïté). Cela revient à considérer comme acquis un patron analogique général de la forme

$$x : f(x) :: y : f(y).$$

Evidemment, dans cet exemple numérique, nous avons au moins 2 options : $f_1(x) = 2 * x$ et $f_2(x) = x + 2$ qui conduisent respectivement à prédire 38 et 21 pour l'élément manquant. Si nous considérons cependant une séquence plus longue, telle que 3, 6, 9, 5, 5, 10, 7, 8, ?, en essayant de trouver un seul patron analogique, on peut observer que $9 = 3 + 6 = f(3, 6)$, $10 = 5 + 5 = f(5, 5)$ et ainsi appliquer f à $(7, 8)$ pour prédire l'élément manquant $7 + 8 = 15$. Cette façon de faire conduit à considérer le même patron que précédemment mais où le type des éléments x et y n'est pas le même que celui de $f(x)$ et $f(y)$, i.e. le domaine de f et son co-domaine sont différents. Ce qui correspond au patron

$$(x_1, x_2) : f(x_1, x_2) :: (y_1, y_2) : f(y_1, y_2).$$

C'est une manière utile pour gérer des éléments de types différents dans une même proportion analogique, et ainsi pouvoir traiter des problèmes plus généraux. Dans notre exemple numérique, nous considérons $(3, 6) : 9 :: (5, 5) : 10$ et devons compléter $(5, 5) : 10 :: (7, 8) : ?$. La fonction qu'il faut appliquer est $f(x, y) = x + y$ de \mathbb{R}^2 dans \mathbb{R} . Nous avons maintenant assez de matériel pour le problème considéré, avec les 3 patrons que nous venons de discuter, cf. Figure 2. Illustrons maintenant l'usage de ces patrons sur des problèmes analogiques à base d'images.

FIGURE 2 – Patrons analogiques généraux

$a b a b$ ou $a a b b$	(1) patron de base
$a f(a) b f(b)$	(2) extension fonctionnelle
$\vec{a} \vec{b} \vec{c} \vec{d}$	(3) extension vectorielle

2.3 Exemples de base

En fait, n'importe quelle séquence de 3 éléments A, B, C (codés par des vecteurs booléens), et devant être complétée avec un élément D pour construire une proportion analogique, est soumise à l'extension vectorielle de la proportion. En fonction de la situation, nous pouvons rencontrer une solution ou un échec. Considérons l'exemple de la Figure 3. Une manière simple de coder le

FIGURE 3 – Une séquence analogique simple



problème est de considérer les prédicats booléens 'comporte un Carré', 'comporte un PointNoir', 'comporte un Triangle', 'comporte un Cercle' et 'comporte une Ellipse' dans cet ordre. Cela conduit au codage donné en Figure 4. En appliquant le processus de résolution compo-

FIGURE 4 – Un codage booléen

a	1	1	0	0	1
b	1	1	0	1	0
c	0	1	1	0	1
d	?	?	?	?	?

sante par composante, nous obtenons $d = (0, 1, 1, 1, 0)$ qui est exactement le codage de la solution attendue. On peut remarquer que nous utilisons toutes les lignes de la table de vérité de l’analogie (à l’exception de la ligne 0000 qui pourrait être facile à obtenir en ajoutant une caractéristique manquante comme ‘comporte un PointBlanc’ par exemple). Ceci n’est pas une règle générale, et nous pouvons facilement construire des séquences où par exemple 0011 et 1100 n’apparaissent pas. Cette approche de base ne peut pas être appliquée à une séquence telle que

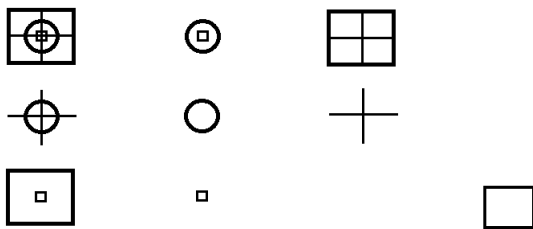
$$A : V :: B : ?$$

Dans ce cas, nous devons considérer l’extension fonctionnelle du patron avec $a : f(a) :: b : f(b)$ où f indique une rotation de 180 degrés. En appliquant ce patron, nous obtenons $? = \text{B}$ comme prévu.¹

3 Une approche pour les tests de Raven

Originellement proposé par John C. Raven en 1936, les matrices progressives de Raven (MPR) [21] constituent une série de tests non verbaux où le participant est mis en présence de matrices 3×3 d’images où il manque l’image en bas à droite, qui doit être choisie parmi une série de 8 réponses candidates. La performance résultante est considérée comme une mesure de la capacité de raisonnement du participant, et en fin de compte de son intelligence générale. En conséquence, c’est un outil largement utilisé dans le champ de l’évaluation de l’intelligence. Un exemple² et sa solution (un grand carré) sont donnés en Figure 5. Ré-

FIGURE 5 – test de Raven et sa solution



soudre un test de MPR est donc essentiellement un exercice visuel et dépend fortement de la représentation de l’espace

1. Par contre, on a “formellement” $A : A \text{ renversé} :: B : B \text{ renversé}$!
 2. Pour protéger la sécurité des tests de Raven, les exemples sont des transpositions des tests réels, qui leur sont parfaitement isomorphes au sens où toutes les relations logiques entre constituants sont préservées, leur apparence graphique pouvant être notablement différente.

et des objets considérés. Chaque test est constitué d’une matrice d’images 3×3 notées ici $pic[i, j]$ où manque la dernière image $pic[3, 3]$ qui doit être choisie dans un ensemble de 8 images candidates.

Nous faisons l’hypothèse que les matrices de Raven peuvent être comprises de la manière suivante, en ce qui concerne les lignes et les colonnes :

$$\forall i \in [1, 2], \exists f \text{ tel que } pic[i, 3] = f(pic[i, 1], pic[i, 2])$$

$$\forall j \in [1, 2], \exists g \text{ tel que } pic[3, j] = g(pic[1, j], pic[2, j])$$

Les deux lignes complètes (resp. colonnes) sont supposées aider à découvrir f (resp. g) et à prédire l’image manquante $pic[3, 3]$ comme $f(pic[3, 1], pic[3, 2])$ (resp. $g(pic[1, 3], pic[2, 3])$).

Tout d’abord, le schéma d’analogie étendue doit être appliqué pour nous dire si $(a, b) : f(a, b) :: (c, d) : f(c, d)$ est valide pour les lignes et $(a, b) : g(a, b) :: (c, d) : g(c, d)$ pour les colonnes, ce qui peut être traduit comme :

$$(pic[1,1], pic[1,2]) : pic[1,3] :: (pic[2,1], pic[2,2]) : pic[2,3]$$

$$(pic[1,1], pic[2,1]) : pic[3,1] :: (pic[1,2], pic[2,2]) : pic[3,2]$$

Ainsi, dans ce cas, nous devons considérer une paire de cellules $(pic[i, 1], pic[i, 2])$ comme le premier élément d’une proportion analogique. La paire $((pic[i, 1], pic[i, 2]), pic[i, 3])$ fournit les deux premiers éléments a et b de la proportion analogique que nous sommes en train de chercher. C’est pourquoi nous avons besoin du deuxième patron pour construire un patron analogique qui relie des éléments de types différents.

3.1 Représentation par caractéristiques

Si les images sont représentées par des vecteurs booléens de caractéristiques, nous pouvons appliquer nos patrons pour essayer de construire l’image manquante. Ceci conduit à un algorithme unique susceptible de résoudre n’importe quelle matrice de Raven. Notons pic la matrice de Raven : celle-ci est une matrice 3×3 où $pic[i, j]$ ($i, j \in [1, 3] \times [1, 3]$) indique l’image dans la ligne i -colonne j . $pic[3, 3]$ n’est pas connu et doit être prédit. Les autres $pic[1, 1], pic[1, 2], pic[1, 3], pic[2, 1], pic[2, 2], pic[2, 3], pic[3, 1]$ et $pic[3, 2]$ sont complètement définis. Chaque $pic[i, j]$ est représenté par un vecteur booléen de dimension n . L’application de notre principe conduit à l’algorithme :

```

Input: Picture array pic[3,3][n]
//pic[3,3] is unknown
for i=1 to n
  //1st row pattern
  OKRow1-1 = (pic[3,1][i]==pic[1,1][i]);
  OKRow1-2 = (pic[3,2][i]==pic[1,2][i]);
  OKRow1= OKRow1-1 & OKRow1-2;
  if OKRow1 then {pic[3,3][i]=pic[1,3][i];}

```

```

//2nd row pattern
else if {
OKRow2-1 = (pic[3,1][i]==pic[2,1][i]);
OKRow2-2 = (pic[3,2][i]==pic[2,2][i]);
OKRow2= OKRow2-1 & OKRow2-2;
if OKRow2 then {pic[3,3][i]=pic[2,3][i];}
//1st column pattern
else if {
OKCol1-1 = (pic[1,3][i]==pic[1,1][i])
OKCol1-2 = (pic[2,3][i]==pic[2,1][i]);
OKCol1= OKCol1-1 & OKCol1-2;
if OKCol1 then {pic[3,3][i]=pic[3,1][i];}
//2nd column pattern
else if {
OKCol2-1 = (pic[1,3][i]==pic[1,2][i]);
OKCol2-2 = (pic[2,3][i]==pic[2,2][i]);
OKCol2= OKCol2-1 & OKCol2-2;
if OKCol2 then {pic[3,3][i]=pic[3,2][i];}
//
} } } } else {error message}
end for
Output:n-dimension Boolean vector pic[3,3]

```

Ce que fait l'algorithme est, que pour chaque caractéristique i , on compare le patron incomplet trouvé dans la troisième ligne (ou colonne) de la matrice avec ceux complets des autres lignes et colonnes, afin de donner à la position $[3, 3]$ la valeur correspondante en $[j, 3]$ dans le cas où on trouve le même patron en ligne j , ou la valeur correspondante en $[3, j]$ si on le trouve en colonne j .

En fait, l'algorithme ci-dessus est en général trop rigide. En effet, on doit pouvoir considérer qu'une matrice de Raven est en accord avec une proportion analogique, sans que cela soit nécessairement exemplifié pour chaque caractéristique considérée individuellement. Dans ce cas, l'algorithme précédent doit être légèrement modifié pour prendre en compte cette option. Nous ne réécrivons pas complètement l'algorithme, mais le changement principal vient du fait que pour résoudre la caractéristique i , nous nous permettons d'utiliser une observation relative à une caractéristique j et d'attribuer à i la solution associée à j , que cela soit en ligne ou en colonne (ceci dans la mesure où il n'existe pas d'association distincte concurrente à cette association). C'est évidemment une extension importante, qui est absolument nécessaire quand nous ne trouvons pas de solution en nous attachant à l'information, en ligne ou en colonne, concernant une caractéristique particulière i .

Dans la Figure 6 un exemple de codage booléen pour deux caractéristiques quelconques est montré. Pour la première, l'algorithme ne trouvera pas dans les deux premières lignes le patron (0, 0), ce qui l'amènera à évaluer les colonnes, trouvant dans la deuxième le patron (0, 1), ce qui permet de donner à la première caractéristique de l'image $[3, 3]$ la même valeur que celle pour l'image $[3, 2]$: 0. Si on analyse la deuxième caractéristique, on pourra consta-

ter qu'on ne peut pas la résoudre directement de la même manière. On ne peut alors compléter le patron (0, 1) qu'en recopiant ce qui se passe sur la première caractéristique, ce qui nous permet ici de lui donner la valeur 0 (pourvu qu'on ne trouve pas de contre-exemple pour le patron incomplet 0, 1, c.-à-d. que le patron 0, 1, 1 n'existe ni en ligne ni en colonne pour une caractéristique quelconque).

FIGURE 6 – Exemple de codage booléen

	1	2	3
1	11	01	00
2	11	10	11
3	00	01	?????

Comme nous pouvons le remarquer, quand il n'y a pas de solution analogique, un message d'erreur est retourné, et on ne peut pas résoudre le test. Comme on a suggéré dans [19], quand un patron analogique n'est pas adéquat, nous pourrions essayer d'en utiliser un autre venant d'une autre proportion, augmentant ainsi notre capacité de résolution, mais ceci est hors du champ de cet article.

3.2 Représentation au niveau des pixels

Evidemment, le codage booléen n'est pas fait automatiquement dans l'approche ci-dessus, et on peut se demander si nous pourrions compter sur un type quelconque de codage automatique. Si nous considérons une image comme une matrice de pixels de dimension $n \times m$ (en considérant seulement des formats non comprimés comme BMP, par exemple), un codage booléen (ou similaire) simple est déjà effectué par le système chargé de capturer les images (e.g. la caméra ou le scanner). Dans ce cas, au lieu de considérer 8 vecteurs construits à la main, nous utilisons des fichiers BMP par exemple. A part éventuellement le fait que nous devons tenir compte des en-têtes de fichiers (qui ne suivent aucun patron de proportion), il n'y a aucune raison de changer notre méthode et notre algorithme de base s'applique sans aucun changement à la partie données du fichier. Clairement, pour que notre algorithme fonctionne correctement, nous devons implicitement supposer que les 8 fichiers ont exactement les mêmes dimensions (ce qui évidemment peut être fait sur des images).

Nous donnons et analysons maintenant les résultats de notre algorithme avec les deux types de représentations.

4 Exemples et résultats

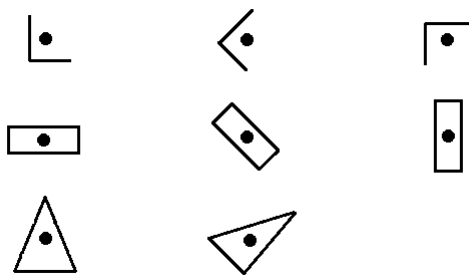
Il y a dans la littérature diverses séries de tests de MPR. Dans ce papier, nous avons considéré la partie "avancée" de ces séries qui sont normalement divisées en une sous-série de 12 tests considérés comme "faciles" et une autre de 36, qui sont normalement dédiés aux adultes avec une intelligence au-dessus de la moyenne. Nous nous concentrons ici

sur la deuxième sous-série de 36 tests généralement considérés comme les plus difficiles à résoudre. Nous avons appliqué notre algorithme à chacun de ces 36 tests, au niveau des caractéristiques globales et au niveau des pixels. Nous discutons des exemples de succès et d'échecs pour les deux niveaux de codage.

4.1 Approche au niveau pixels

Commençons avec l'approche au niveau pixels où nous mettons l'accent sur le dispositif de traitement d'images pour fournir le codage. Quand nous appliquons notre algorithme aux 36 tests de la série, nous obtenons 16 résultats positifs (où la solution est construite), laissant donc 20 tests non résolus. Parmi les tests que nous résolvons avec cette approche, il y a le test Raven numéro 12 déjà considéré. Un exemple d'échec est le test 10 de la Figure 7 que nous discutons maintenant. La solution fournie par notre algo-

FIGURE 7 – Raven test 10 modifié



ritme en utilisant une représentation par pixels est montrée sur la Figure 8 et la solution correcte est montrée sur la Figure 9, remarquant que notre solution garde seulement ici la caractéristique qui apparaît dans toutes les images : le "point noir". Notre algorithme est incapable de fournir les lignes autour de ce point nécessaires pour obtenir une solution correcte.

FIGURE 8 –Raven test10 modifié,ce que l'algorithme donne



FIGURE 9 – Raven test 10 modifié : la vraie solution



Cette situation ne doit pas apparaître comme une surprise, et nous ne pouvons pas espérer capturer toute l'information contenue dans les matrices avec une représentation de si bas niveau (pixels). Généralement, nous échouons quand la variation d'un objet à travers les images est lié à une modification des formes. Il est malgré tout remarquable que

cette méthode soit suffisante pour résoudre 16 tests. Considérons maintenant l'approche avec un niveau de représentation plus haut où nous considérons encore un codage booléen mais qui est maintenant fait manuellement.

4.2 Approche par caractéristiques globales

Quand on utilise l'approche booléenne appliquée à des caractéristiques globales, nous pouvons résoudre, en plus de la liste précédente, 16 nouveaux tests, débouchant sur un total de 32 tests parmi les 36. Nous examinons le traitement des 4 derniers cas ensuite. Considérons ici 2 exemples qui sont relativement difficiles parmi les MPR. Tout d'abord, nous revenons sur le test Raven 10, que nous avons échoué à résoudre avec l'approche par pixels. Nous pouvons utiliser le codage booléen suivant qui donne la solution exacte. Étant donné que le point central est constant dans les 8 images, il n'est pas utile de le coder ici, car nous savons que ce point apparaîtra dans l'image que nous voulons construire (et c'est exactement ce qui se passe avec l'approche par pixels). Par ailleurs, nous avons 6 autres caractéristiques décrivant le problème, ce qui nous conduit à une représentation par un vecteur booléen de dimension 6 :

- 3 caractéristiques décrivant la forme des figures : *formeL*, *rectangle* et *triangle*,
- 3 caractéristiques décrivant la position des figures : *position1*, *position2* et *position3*. La position 2 est la rotation de 45 degrés à droite de la position 1, et la position 3 est la rotation de 45 degrés à droite de la position 2.

FIGURE 10 – Codage de haut niveau du test Raven 10

	1	2	3
1	100100	100010	100001
2	010100	010010	010001
3	001100	001010	??????

Le codage complet est donné en Figure 10. Avec une analyse verticale, nous obtenons pour la dernière image le codage 001001. Nous devons remarquer que verticalement nous ne pouvons pas compléter le patron 11 pour le dernier attribut. Nous devons alors changer la colonne cible pour trouver un contexte similaire dans la première colonne (par exemple) et la sous-colonne 4, ce qui nous donne la valeur finale : 1. Considérons maintenant un deuxième exemple, le test Raven 27 que nous fournissons avec sa solution dans les Figures 11 et 12, respectivement. Un codage booléen de ce test peut être basé sur les caractéristiques

- pour l'existence ou non de lignes dans les parties considérées : *carréExtérieur*, *carréCentral*, *carréIntérieur*,
- pour la direction des lignes : *vertical*, *diagonal* et *horizontal*,

ce qui nous amène à nouveau à un vecteur booléen de dimension 6. Nous montrons le codage complet en Figure 13. En résolvant cette matrice avec notre algorithme, nous obtenons le codage 100010 qui est le codage exact de l'image

solution. Pour conclure cette partie, récapitulons nos résultats dans le tableau de la Figure 14.

FIGURE 11 – Test de Raven 27 modifié

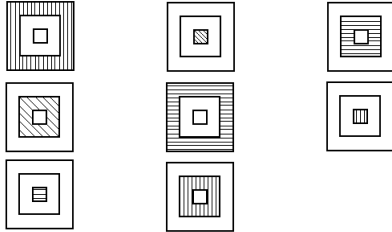


FIGURE 12 – Solution du test de Raven 27 modifié



FIGURE 13 – Un codage booléen du test Raven 27 modifié

	1	2	3
1	100100	001010	010001
2	010010	100001	001100
3	001001	010100	??????

FIGURE 14 – Les résultats obtenus

	approche "pixel"	approche "haut niveau"
résolus	16	16+16
non résolus	20	4
total	36	36

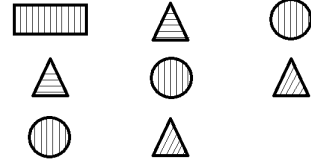
4.3 En utilisant l'algèbre de Post

Parmi les cas d'échec, certains sont dus au fait qu'une figure, ou du moins une de ses caractéristiques globales, est en quelque sorte soumise à une fonction de rotation sur l'ensemble des valeurs possibles pour cet attribut, comme par exemple en Figure 15 : on observe un attribut pouvant prendre 4 valeurs R, V, B et J et on attend comme solution R , mais le calcul Booléen ne donne pas de solution à cause de contradictions. Dans ce genre de cas, on peut cependant utiliser un codage basé sur une algèbre de Post, comme évoquée dans [17]. Dans cette approche, un opérateur σ permet de définir une négation (cyclique), tout en conservant la définition de la proportion analogique mais avec un sens étendu. Etant donné une série ordonnée (l'ordre peut être choisi arbitrairement) d'items $\{a_1, \dots, a_n\}$, on définit $\sigma(a_i) = a_{i+1}$, pour $i \in [1, n-1]$, et $\sigma(a_n) = a_1$. La négation d'un élément a est simplement $\sigma(a)$. Le schéma étendu $a : f(a) :: b : f(b)$ devient $a : \sigma^k(a) :: b : \sigma^k(b)$ et on voit que l'exemple simplifié de la Figure 15 se résout parfaitement via ce codage. Considérons un exemple plus complet (un test MPR), celui de la Figure 16, où l'on utilise les attributs *rectangle*, *circle*, *triangle* pour décrire les formes et les attributs *vertical*, *horizontal* et *diagonal* pour décrire les lignes en ordonnant de la manière suivante :

FIGURE 15 – Un cas typique d'échec du codage Booléen

R	V	B		1000	0100	0010
V	B	J	->	0100	0010	0001
B	J	?		0010	0001	?

FIGURE 16 – Exemple codé "à la Post"



a_1 : rectangle a_2 : triangle a_3 : cercle
 b_1 : vertical b_2 : horizontal b_3 : diagonal

Ce qui nous mène au codage de la Figure 17. Pour pou-

FIGURE 17 – Le codage avec l'algèbre de Post

$a_1 b_1$	$a_2 b_2$	$a_3 b_3$
$a_2 b_2$	$a_3 b_1$	$a_2 b_3$
$a_3 b_1$	$a_2 b_3$	$x y$

voir bénéficier de ce codage cyclique, il faut de plus utiliser l'analogie sous forme de proportions continues, qui sont de la forme $A : B :: B : C$, pour exprimer (en ligne ou en colonne) que si la cellule C suit la cellule B qui suit A , alors d'une certaine façon "A est à B ce que B est à C". Ainsi, pour résoudre selon l'attribut a , il faut donc évaluer le patron $a_3 : a_2 :: a_2 : a_1$ selon la logique de Post. Puisque $a_2 = \sigma^2(a_3)$, on obtient $x = \sigma^2(a_2) = a_1$. Le même raisonnement s'applique pour l'attribut b et on obtient ici $y = b_2$. Or $a_1 b_2$, est exactement le codage de la solution attendue (Figure 18). Ceci permet de résoudre 2 tests de Raven de plus.

FIGURE 18 – Solution de l'exemple Post



4.4 Les deux échecs restants

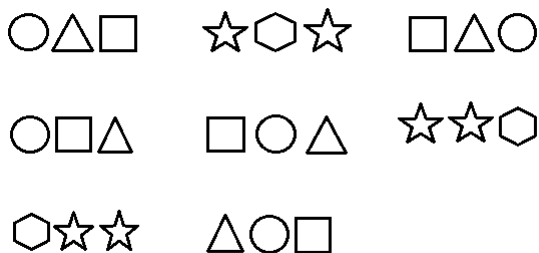
L'analogie reste donc impuissante à résoudre 2 tests de Raven. Dans un cas, l'approche analogique, en utilisant les codages cyclique ou booléen, trouve des contradictions, c'-à-d. deux réponses concurrentes pour une même caractéristique. Pour ce cas, on trouve cependant, avec le codage booléen, les bonnes réponses pour toutes les caractéristiques sauf une, et pour la situation contradictoire, une des options est la bonne, tandis que l'autre conduit à une solution possible qui ne figure pas dans les 8 proposées, ce qui permettrait de conclure par élimination, mais l'approche n'est plus alors entièrement constructive.

Sur la Figure 19 on peut voir le dernier cas que notre approche n'a pas pu résoudre. Dans cet exemple, il faut déterminer les 3 figures qui doivent apparaître dans la réponse, ainsi que la position de chacune. On peut remarquer qu'il y a des situations qui sont constantes en ligne ou en colonne :

(a) Il y a deux étoiles dans chaque ligne et chaque colonne ; (b) il y a deux cercles dans chaque ligne et chaque colonne, et de plus en colonne, ils sont dans la même position ; (c) il y a deux triangles par ligne et par colonne, et de plus en ligne ils sont dans la même position ; (d) il y a un hexagone par ligne et par colonne.

Avec un codage booléen, en utilisant les noms des figures comme attributs (*cercle*, *triangle*, *carré*, *étoile*, *hexagone*) sans prendre en compte ni sa position ni son nombre d'apparition dans chaque image, nous savons que l'image de la réponse doit contenir un cercle, un triangle et un carré. C'est en essayant de connaître la position de chacune de ces figures que notre approche n'arrive pas à déterminer la réponse puisque c'est dans ce cas qu'il faut évaluer les lignes et colonnes d'une manière générale en fonction des contraintes mentionnées ci-dessus : La contrainte (b) nous dit que le cercle maintient sa position parmi les colonnes, et comme dans l'image [3, 1] le cercle est à droite, il doit avoir la même position dans la réponse. En évaluant la contrainte (c) on sait que le triangle doit être à gauche de l'image et cela conduit à déduire que le carré doit apparaître au centre de l'image. La réponse est donc "triangle carré cercle".

FIGURE 19 – Exemple d'échec



Ceci montre que la résolution complète de l'exemple, au delà de la recopie met en jeu des contraintes entre caractéristiques pour conclure ici que la position (centrale) restante ne peut être occupée que par l'élément restant (le carré), qui ne semblent pas pouvoir être facilement prises en compte par une pure approche analogique. On peut aussi remarquer sur la Figure 19 que la solution correspond à la seule permutation des trois éléments (triangle, carré, cercle) qui manque, les trois autres cellules correspondant à toutes les permutations possibles d'un hexagone et de deux étoiles. Cette vue est au delà de l'analogie.

5 Approche par "Structure Mapping"

Vouloir résoudre des proportions analogiques liées à des diagrammes ou des figures géométriques n'est pas un problème nouveau : nous avons rappelé en introduction les travaux pionniers de [2][3]. A notre connaissance, pour ce qui est des tests de Raven, la première approche computationnelle fut développée par Carpenter et col. [1]. Basée sur des études psychologiques, leur approche utilise le fait que les humains résolvent ces tests de la manière suivante :

- En observant les 2 premières lignes de la matrice (sans considérer les colonnes) ;
- Ensuite en générant des hypothèses de règles à partir de leurs observations : Dans [1], un ensemble de 6 règles possiblement applicables a été ainsi retenu ;
- Enfin, en établissant une liaison entre les deux premières lignes et la dernière en termes de similitude et de dissemblance pour savoir comment appliquer à cette dernière les règles apprises à l'étape précédente.

Les règles utilisées par le programme sont choisies parmi les 6 règles retenues (chaque règle décrit comment un groupe d'objets varie à travers les trois images d'une même ligne). A partir de ces observations, [1] présente deux programmes, FAIRAVEN et BETTERAVEN. Tous deux utilisent des représentations faites à la main et les règles pour déterminer la réponse sont choisies parmi les 6 déjà codées dans le système. BETTERAVEN était capable d'égaliser la performance des meilleurs participants dans les tests de MPR. L'information venant des colonnes ne semble pas être utilisée dans ce processus global. L'approche de [1] ne fait aucune référence au raisonnement analogique.

Récemment, Lovett, Forbus, et Usher [13] ont souligné qu'au delà de l'appréhension de figures spatiales, la résolution des MPR paraît utiliser une autre capacité cognitive très importante, à savoir celle de mettre en œuvre des analogies. En partant de cette supposition, les auteurs de [13] combinent des représentations spatiales qualitatives avec une comparaison analogique, ce qui a conduit à un modèle beaucoup plus puissant que le précédent. L'approche est basée sur un modèle d'analogie bien connu, la *Structure Mapping Theory (SMT)* de Gentner [8], et son implémentation utilise le *Structure Mapping Engine (SME)* [5].

SME construit des analogies à la manière dont les humains sont supposés le faire, c'est à dire en construisant un *mapping* des connaissances d'un domaine vers un autre. SME prend en entrée les descriptions du couple de situations à comparer, comprenant des entités (objets), des prédicats (qui relient les entités entre elles), et enfin des relations qui peuvent regrouper prédicats et entités. Chacun de ces couples formés constitue une expression. Ces ensembles d'expressions forment des graphes acycliques, dont les noeuds sont les entités et la racine représente une relation d'ordre supérieure qui relie les éléments du graphe.

SME commence par créer des *match hypothesis* entre les deux descriptions, où les expressions sont assemblées par paires, sur la base de l'identité de prédicats ou d'entités. Ensuite, chaque *match hypothesis* est comparée avec toutes les autres pour déterminer les incompatibilités entre elles, afin de pouvoir combiner toutes les *match hypothesis* susceptibles de former des ensembles maximaux cohérents. La détermination de conflit entre couples d'expressions est basée sur des règles d'appariement "un-à-un", en convenant que chaque expression ne peut pas avoir un *match* avec deux expressions différentes. Cette règle dimi-

nue la capacité de prendre en compte des expressions ou objets qui font partie de relations dans des contextes différents en ayant un rôle dans chacun [4].

Pour chaque ensemble maximal obtenu, SME prend aussi en compte dans leur évaluation, des expressions qui apparaissent seulement dans une des deux descriptions (donc en dehors de l'ensemble maximal), mais qui ont cependant des sous-arguments en commun avec les expressions de l'ensemble. Elles sont appelées "candidats pour l'inférence" (car possiblement vraies dans la description dont elles sont absentes). SME évalue chacun de ces ensembles maximaux en calculant un score qui dépend en particulier du nombre d'identités entre prédicats des deux descriptions, en prenant aussi en compte la structure de graphe reflétant les liens entre entités et prédicats et entre prédicats. SME fournit en sortie les ensembles maximaux, avec leur candidats pour l'inférence et leur score. On a cependant pu mettre en doute que la reconnaissance des analogies chez l'humain nécessite l'identification de telles mises en correspondance et leur évaluation [22].

Pour résoudre les MPR, les représentations spatiales des images sont obtenues en utilisant CogSketch [6], un système de reconnaissance qui prend en entrée une esquisse (dessinée, et segmentée par objets par l'utilisateur), et génère une représentation qualitative de ces objets (ou de ses arêtes) ou de groupes d'objets et de leurs relations (position relative, topologie, etc). Par exemple, CogSketch peut dire quels objets dans une esquisse sont côte-à-côte, lesquels se chevauchent, ou sont à l'intérieur d'un autre.

Ensuite, CogSketch et SME sont utilisés pour comparer deux objets dans une esquisse, mais en utilisant leurs représentations au niveau des arêtes, pour déterminer s'il y a une transformation (rotation, taille), ou une déformation (transformation totale de la forme) entre ces deux objets. Avec cette information, les objets de forme équivalente, sont regroupés ensemble.

Afin de sélectionner la réponse correcte parmi les choix possibles, Lovett et col. [13] utilisent SME à nouveau en combinant quatre stratégies. L'approche commence par évaluer les deux premières lignes du test pour générer quelques règles, appelées patrons de variation. Ensuite, les patrons de variation trouvés pour chaque ligne sont comparés pour calculer un score de similarité, qui confronté à un seuil permet de déterminer leur validité. Si les patrons de variation sont considérés suffisamment similaires, le programme utilise les 8 réponses possibles du test en les évaluant avec le même patron de variation, et choisit la réponse avec le plus grand score. Ce modèle a été évalué sur les 48 problèmes des sections B-E du test Standard de Matrices progressives, et en a résolu 44. Lovett et col. [13] expliquent que les 4 problèmes qui n'ont pas pu être résolus sont parmi les six les plus durs pour les humains, et sont les problèmes qui ont le taux de résolution le plus bas.

On peut donc observer deux grandes différences entre l'approche de Lovett et col. [13] et notre solution :

- [13] utilise plusieurs règles combinées avec 4 stratégies pour résoudre les tests. Notre approche apparaît plus simple avec un principe uniforme de recopie appliqué quel que soit le problème.
- Notre approche fournit un processus constructif basé sur la proportion analogique, en calculant pas à pas l'image solution, sans la choisir parmi la série des solutions candidates.

6 Conclusion

Nous avons présenté une approche basée sur un principe unique qui nous permet de résoudre presque tous les problèmes de matrices progressives de Raven. Les problèmes restant mettent en jeu des mécanismes de propagation de contraintes qui paraissent hors de portée du pur raisonnement analogique. Il est à noter que l'approche est basée sur un principe unique, qui découle immédiatement de la modélisation logique des proportions analogiques, et en fait ici, d'une restriction de cette modélisation logique aux cas de l'identité dans le changement ou dans l'absence de changement. Cela équivaut à recopier ce qui peut être observé, et à le recopier éventuellement dans le contexte d'autres caractéristiques s'il n'y a aucune autre observation contradictoire empêchant cette recopie. De plus, dans un certain nombre de cas, l'approche peut être appliquée directement au niveau pixels.

Notre approche, au contraire de Forbus et col. [4, 13] et Carpenter et col. [1], n'est basée sur aucune étude cognitive. Même si le fait que notre méthode utilise simplement une recopie des données puisse faire douter de la pertinence de tels tests pour mesurer l'intelligence humaine, il faut remarquer cependant que le processus humain pour les résoudre est nécessairement plus riche qu'un simple principe de recopie, malgré tout peut-être aussi à l'oeuvre chez l'humain. En effet, on ne peut pas juste caractériser l'intelligence comme une question de recopie de solutions, puisque le processus implique aussi la recherche et la sélection de l'information à recopier. L'identification de cette information peut d'ailleurs être plus ou moins compliquée selon la complexité apparente des formes présentées dans les tests (des tests isomorphes en termes de structures logiques peuvent sans doute apparaître de niveaux de difficulté différents pour un humain). Au plan de l'implémentation, la différence avec l'humain est probablement plus flagrante : En premier lieu, effectuer une analyse similaire à celle réalisée dans l'approche par pixels est impossible pour un humain étant donné la quantité de données à prendre en compte, et le codage systématique par caractéristiques, même s'il est parfaitement réalisable à la main, peut nécessiter plus de temps que celui dont les candidats disposent pour les tests.

Il est intéressant de remarquer que les proportions ana-

logiques, ont donné des résultats prometteurs en classification [14, 20], en général meilleurs que les résultats obtenus avec la méthode des k plus proches voisins. Etant donné une série d'exemples, chacun décrit au moyen d'un vecteur de caractéristiques (c_1, \dots, c_n) , chaque vecteur étant associé à une classe cl , un nouveau vecteur est classé au moyen d'une prédiction basée sur un triplet d'exemples pour lesquels une proportion analogique est valide avec ce nouveau vecteur, pour un nombre maximum de caractéristiques. Ce qui ouvre aux proportions analogiques un champ d'application autrement plus large que les tests de QI !

Références

- [1] P. A. Carpenter, M. A. Just, and P. Shell. What one intelligence test measures : A theoretical account of the processing in the raven progressive matrices test. *Psychological Review*, 97(3) :404–431, 1990.
- [2] T. G. Evans. A heuristic program to solve geometry-analogy problems. In *Proc. A.F.I.P. Spring Joint Computer Conf.*, volume 25, pages 5–16, 1964.
- [3] T. G. Evans. A program for the solution of a class of geometric-analogy intelligence-test questions. In Marvin L. Minsky, editor, *Semantic Information Processing*, pages 271–353. MIT Press, Cambridge, Massachusetts, 1968.
- [4] B. Falkenhainer. Analogical interpretation in context. In *Proc. 12th Annual Conf. of The Cognitive Science Society*, pages 69–76. Lawrence Erlbaum, 1990.
- [5] B. Falkenhainer, K. D. Forbus, and D. Gentner. The structure-mapping engine : algorithm and examples. *Artif. Intell.*, 41(1) :1–63, 1989.
- [6] K. Forbus, J. Usher, A. Lovett, K. Lockwood, and J. Wetzel. Cogsketch : Sketch understanding for cognitive science research and for education. *Topics in Cognitive Science*, pages 648–666, 2011.
- [7] R. M. French. The computational modeling of analogy-making. *Trends in Cognitive Sciences*, 6(5) :200 – 205, 2002.
- [8] D. Gentner. Structure-mapping : A theoretical framework for analogy. *Cog. Sci.*, 7(2) :155–170, 1983.
- [9] D. Gentner. The Mechanisms of Analogical Learning. In *Similarity and Analogical Reasoning*, pages 197–241. Cambridge University Press, 1989.
- [10] D. Hofstadter and M. Mitchell. The Copycat project : A model of mental fluidity and analogy-making. In D. Hofstadter and The Fluid Analogies Research Group, editors, *Fluid Concepts and Creative Analogies : Computer Models of the Fundamental Mechanisms of Thought*, pages 205–267, New York, NY, 1995. Basic Books.
- [11] J. Holyoak and P. Thagard. Analogical mapping by constraint satisfaction. *Cog. Sci.*, 13 :295–355, 1989.
- [12] J. E. Hummel and K. J. Holyoak. Distributed representations of structure : a theory of analogical access and mapping. *Psy. Rev.*, 104(3) :427–466, 1997.
- [13] A. Lovett, K. Forbus, and J. Usher. A structure-mapping model of Raven's progressive matrices. In *Proc. of the 32nd Annual Conference of the Cognitive Science Society, Portland, OR*, 2010.
- [14] L. Miclet, S. Bayouhd, and A. Delhay. Analogical dissimilarity : definition, algorithms and two experiments in machine learning. *JAIR*, 32 :793–824, 2008.
- [15] L. Miclet and H. Prade. Handling analogical proportions in classical logic and fuzzy logics settings. In *Proc. 10th Eur. Conf. on Symb. and Quantit. Appr. to Reas. with Uncert. (ECSQARU'09), Verona*, pages 638–650. Springer, LNCS 5590, 2009.
- [16] H. Prade and G. Richard. Logical proportions - typology and roadmap. In E. Hüllermeier, R. Kruse, and F. Hoffmann, editors, *Computational Intelligence for Knowledge-Based Systems Design : Proc. 13th Inter. Conf. on Inform. Process. and Management of Uncertainty (IPMU'10), Dortmund, June 28 - July 2*, volume 6178 of LNCS, pages 757–767. Springer, 2010.
- [17] H. Prade and G. Richard. Multiple-valued logic interpretations of analogical, reverse analogical, and paralogical proportions. In *Proc. 40th IEEE Int. Symp. on Multiple-Valued Logic (ISMVL'10), Barcelona*, pages 258–263, 2010.
- [18] H. Prade and G. Richard. Reasoning with logical proportions. In *Proc. 12th Inter. Conf. on Princip. of Knowledge Represent. and Reas. (KR 2010), Toronto, May 9-13, 2010 (F. Z. Lin, U. Sattler, M. Truszczynski, eds.)*, pages 545–555. AAAI Press, 2010.
- [19] H. Prade and G. Richard. Analogy-making for solving IQ tests : A logical view. In *Proc. 19th Inter. Conf. on Case-Based Reas., London, 12-15 Sept.*, volume 6880 of LNAI, pages 241–250. Springer, 2011.
- [20] H. Prade, G. Richard, and B. Yao. Enforcing regularity by means of analogy-related proportions-a new approach to classification. *International Journal of Computer Information Systems and Industrial Management Applications*, 4 :648–658, 2012.
- [21] J. Raven. The Raven's progressive matrices : Change and stability over culture and time. *Cognitive Psychology*, 41 :1 – 48, 2000.
- [22] T. Ripoll and J. Eynard. La détection d'analogies potentielles sans mapping est-elle possible ? In *Actes Jour. Nat. sur les Modèles de Raisonnement (JNMR'03), Paris, 27-28 nov.*, pages 217–230, 2003.
- [23] P. H. Winston. Learning and reasoning by analogy. *Com. ACM*, 23, 689-703, 1980.