Reconnaissance de formules mathématiques Arabes par un système dirigé par la syntaxe

Afef Kacem, Kawther Khazri, Abdel Belaid

Reconnaissance de formules mathématiques Arabes par un système dirigé par la syntaxe

Afef Kacem*,* — Kaouther Khazri* — Abdel Belaïd*,**

**UTIC-ESSTT- Tunisie*
5 Avenue Taha Hussein, BP 56 Bab Mnara, Tunis,Tunisie
Afef.kacem@esstt.rnu.tn, kaouther.khazri@utic.rnu.tn, abelaid@loria.fr
*** LORIA-CNRS-France*

RÉSUMÉ. *L'objet de cette contribution est de présenter un système dirigé par la syntaxe qui reconnaît des formules mathématiques Arabes et retourne les résultats de la reconnaissance dans le format MathML. Un ensemble de règles de remplacement est défini par une grammaire de coordonnées pour analyser des formules mathématiques Arabes. Cette grammaire est employée en s'appuyant sur la reconnaissance de symboles et l'analyse de leur arrangement spatial. Nous avons utilisé les k plus proches voisins pour reconnaître des symboles mathématiques Arabes et un analyseur syntaxique à la fois descendant et ascendant qui repose sur la dominance d'opérateurs pour diviser récursivement la formule en sous formules plus simples. Dans le système proposé, les modules de la reconnaissance des symboles et de l'analyse structurelle s'interagissent d'une manière étroite. Il est ainsi possible d'utiliser des informations structurelles pour aider à deviner les symboles ambigus ou en confusion. Ce système de reconnaissance, dirigé par la syntaxe, a été démontré avec succès sur plusieurs types de formules extraites de différents documents scientifiques Arabes.*

ABSTRACT. *This work presents a syntax-directed system that recognizes Arabic mathematical formulas and outputs the recognition results in MathML format. The proposal system employed a coordinate grammar with emphasis on symbol recognizer as well as symbol arrangement analysis for parsing formulas. The system used the K nearest neighbor method with Hu moments to recognize Arabic mathematical symbols. A top-down and bottom-up parsing scheme, based on operator dominance, is then used to analyze the formula structure. In our system, the recognition and parsing modules interact closely. Thus, it can use structural information to help guess about symbol identities. The syntax-directed recognition system, described here, has been successfully demonstrated) in many types of formulas found in various Arabic documents in science and engineering disciplines.*

MOTS-CLÉS : *Formule mathématique arabe, Symbole, Reconnaissance, K-ppv, Moments de Hu, Analyse descendante-ascendante, Grammaire de coordonnées, MathML.*

KEYWORDS: *Arabic Mathematical formula, symbol, recognition, K-nn, Hu moments, Top-Down and Bottom-Up analysis, coordinate grammar, MathML.*

1. Introduction

Sans être un sujet d'investigations indépendant, la notation mathématique a toujours fait l'objet d'une réflexion profonde et continuelle. La reconnaissance de notations mathématiques est un problème bien connu et l'objet de beaucoup de travaux de recherche. Pour les formules mathématiques Arabes, peu de travaux existent. L'un des grands problèmes qui peut gêner la reconnaissance de l'écriture mathématique arabe vient de l'incohérence de direction dans les flux d'écriture. Ceci est facile à constater dans le style occidental où le texte est naturellement écrit de droite à gauche, par contre le flux mathématique est dans la direction opposée. Ce phénomène, connu sous le nom de bidirectionnalité, introduit une série de difficultés supplémentaires à l'analyse de formules en particulier dans le cas de la reconnaissance hors ligne. Considérons par exemple la simple formule suivante $p<q$. Si telle formule était écrite de gauche à droite, le sens mathématique serait *p inférieur à q*, mais si elle était introduite de la droite vers la gauche, la même expression va signifier *q supérieur à p* qui est une interprétation erronée du contenu mathématique. Évidement, cet exemple démontre que l'interprétation dépend de la direction dans laquelle la formule mathématique est écrite. Au-delà de ces problèmes majeurs, la notation arabe pose d'autres problématiques. Il s'agit notamment de traitement d'une vaste collection de nouveaux glyphes, y compris les alphabets arabes de base ou les extensions de ces alphabets, en tenant compte des formes pointillés ou non pointillés des lettres et d'autres séries de chiffres. En outre, nous devons reconnaître tout un ensemble de symboles supplémentaires et les ligatures pour certaines fonctions et opérateurs.

L'objectif ultime de ce travail est d'arriver à reconnaître automatiquement des formules mathématiques écrites en Arabe, motivé principalement par l'importance des informations que peut contenir ce type de notation et l'utilité de rendre ces informations accessibles à tout le monde. Cet article porte sur la méthodologie proposée pour atteindre ce but par combinaison des résultats de reconnaissance des symboles mathématiques, contenus dans les formules, et leur analyse structurelle.

Ce papier est organisé en cinq sections. La première section permettra de prendre connaissance des difficultés liées à la reconnaissance de la notation mathématique Arabe. Nous faisons, en section 3, un survol des techniques de reconnaissance présentées dans la littérature. L'architecture ainsi que la méthode mise en œuvre seront décrites dans la section 4. Une première sous section est consacrée à la reconnaissance des symboles mathématiques : étape importante à l'analyse des formules. Nous nous pencherons, en sous section deux, sur l'exposé de la problématique d'analyse structurelle des formules. Une dernière section aura pour rôle de rendre compte, discuter des résultats obtenus et évaluer la qualité de la reconnaissance. Ce papier sera clôturé par une conclusion générale où sont rappelés la problématique, les résultats escomptées et ceux obtenus ainsi que les horizons ou les possibilités d'amélioration futures qui pourraient être réalisées.

2. Notations Mathématiques Arabe

Actuellement, dans les pays arabes, l'écriture de la composante symbolique des textes mathématiques prend différentes formes qu'on peut répartir grossièrement en deux grandes options [Khaled 92]:

— à l'occidentale comme dans les textes mathématiques en anglais ou bien comme ceux en français. Les symboles sont alors empruntés principalement à l'une ou à l'autre de ces deux langues, selon l'importance de l'influence culturelle. La direction de l'écriture des expressions symboliques suit également celle de la langue d'origine, de gauche à droite, en opposition de l'écriture du texte qui suit le sens de l'écriture de la langue naturelle arabe, de droite à gauche (Voir figure 1 a) ;

— à l'orientale où des symboles spécifiques sont alors d'usage. L'écriture des expressions symboliques suit le sens de l'écriture de la langue naturelle, de droite à gauche. Ce sont ces systèmes d'écriture, en conformité avec les normes et les conventions adoptées, qui sont en vigueur dans les manuels scolaires des pays du Moyen Orient, en Libye, en Algérie, etc. Ces symboles étaient également d'usage dans les manuels marocains avant l'adoption du système symbolique français pour la composition du document mathématique arabe (Voir figure 1 b et c).

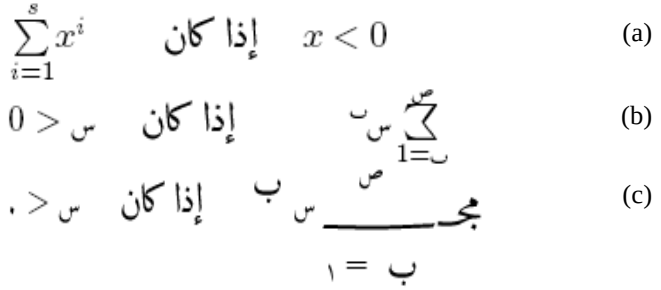


Figure 1. *(a) Style occidental, (b) et (c) styles orientaux*

Les formules mathématiques Arabes nécessitent un grand nombre et une grande diversité de signes [Lazrek 01]:

— Des lettres de plusieurs alphabets (Arabe, Latin, etc.), en plusieurs styles (Naskh, Koufi, etc. pour l'arabe, romain, italique, penché, calligraphique, etc. pour l'alphabet latin). Ces alphabets peuvent figurer en minuscule, majuscule, gras, etc. Les majuscules n'ont pas d'analogue en écriture Arabe.Les lettres utilisées dans les expressions mathématiques Arabes épousent leur forme isolée, en initial de mot avec ou sans queue. Ces lettres sont généralement privées des points ou signes diacritiques, utilisés pour marquer les voyelles. Cela restreint le nombre des lettres arabes disponibles pour composer des symboles. Ces lettres sont utilisées dans l'écriture des variables et des constantes mais pas pour les abréviations des fonctions ;

— Des accents et des délimiteurs en plusieurs formes et tailles ;

— Des chiffres en plusieurs formes : Des chiffres du Maghreb (9, 8, 7, 6, 5, 4, 3, 2, 1, 0) et du Machrek (٠,١,٢,٣,٤,٥,٦,٧,٨,٩) ;

— Des signes de ponctuation en plusieurs formes mais aussi la virgule latine. La virgule Arabe, orientée vers le haut, est utilisée par exemple dans un couple de termes (ex. : (س،ص)). La virgule latine, orientée vers le bas, est utilisée par exemple dans un nombre décimal en notation française (ex. : 3,14).

— Des signes d'opérations arithmétiques, relationnels, ensemblistes, etc. ;

— D'autres symboles spécifiques en plusieurs formes et tailles. La forme des symboles est soit les mêmes symboles que ceux utilisés en roman d'usage courant (ex. +, -), soit les mêmes symboles moyennant une inversion du sens (ex. < et >, ∈ et ɜ), soit les mêmes symboles moyennant une symétrie par rapport à l'axe vertical au milieu (ex. ∑ et √ et). La taille des symboles extensibles est fonction du contexte. Elle peut dépendre de la taille de l'expression couverte (ex. Symbole racine) ou de sa position, en indice ou en exposant ;

— Le texte en langue naturelle et la formule mathématique s'interpénètrent et interagissent mutuellement via des mots de liaisons comme إذا كان , مع ,غير ذلك ,حيث (voir figure 1).

3. Etat de l'art

Une étude approfondie de la bibliographie concernant la reconnaissance de la notation mathématique en général met en évidence l'engouement récent pour ce domaine de recherche. La reconnaissance d'une formule mathématique, quelle soit manuscrite ou typographiée, latine ou arabe, consiste typiquement en deux étapes majeures : une étape de reconnaissance de symboles et de caractères et une étape d'analyse structurelle de la formule. Pour la reconnaissance des symboles, on a utilisé les Modèles de Markov Cachés (HMM) dans [Koschinski et al. 95] et [Winkler et al. 95]. On a combiné aussi entre les HMMs et les réseaux de neurones dans [Kosmala et al. 99]. Dans un précédent travail [Kacem et al 01], nous avons adopté une méthode à base de la logique floue pour étiqueter certains symboles mathématiques. Plus tard, on a proposé, dans [Topia et R. Rojas 03], une Machine à Vecteurs de Support (SVM). Dans [Garain 03], on a testé la combinaison entre les HMMs et la mathode d'appariement des modèles. La plupart des chercheurs effectuent la reconnaissance en ligne des symboles mathématiques latins et s'intéressent uniquement à un sous-ensemble de symboles vu le grand nombre de symboles qui peuvent être utilisés dans les expressions mathématiques.

Selon une étude, faite par [Blosteïn et Gravec 97], les méthodes proposées pour la reconnaissance de formules mathématiques latines ne sont ni robustes, ni efficaces, ni génériques. Elles ne sont capables de reconnaître qu'un certain type de formules, la plupart du temps des équations. Certaines d'entre elles s'intéressent uniquement à l'analyse structurelle des formules en admettant que leurs symboles mathématiques sont parfaitement reconnus. Pour celles qui proposent de reconnaître les symboles, certaines difficultés ne sont toujours pas résolues : caractères se

touchant, petits symboles, etc. Les résultats obtenus sont insuffisants dans le cadre d'une utilisation industrielle. Il est en effet nécessaire, d'une part d'obtenir de meilleurs taux de reconnaissance, et d'autre part d'éviter une relecture fastidieuse et coûteuse de l'ensemble des formules. Dans une étude antérieure [Kacem et al 01], nous avons constaté que la plupart des méthodes d'analyse structurelle des formules mathématiques sont basées sur des formes de syntaxe définies explicitement ou implicitement. En effet, les formules ont une syntaxe bien définie et forment ainsi un excellent domaine pour l'application des techniques de reconnaissance dirigées par la syntaxe. Les méthodes syntaxiques s'avèrent utiles et pratiques mais se basent sur une analyse syntaxique qui nécessite une définition précise de la syntaxe de toutes les formules, ce qui n'est pas évident. De plus, elles sont coûteuses en temps de calcul. Plusieurs systèmes de reconnaissance des formules mathématiques obtiennent la structure des formules sans analyse. A la place, quelques règles codées procéduralement ou des techniques d'analyse assez variées ont été utilisées.

4. Système proposé

Le système fonctionne suivant le schéma de la figure 3. Il repose sur deux étapes majeures: la reconnaissance des symboles et l'analyse structurelle. La première étape convertit l'image de la formule en un ensemble de symboles. La seconde étape analyse l'arrangement spatial des afin de rétablir l'information contenue dans la formule.

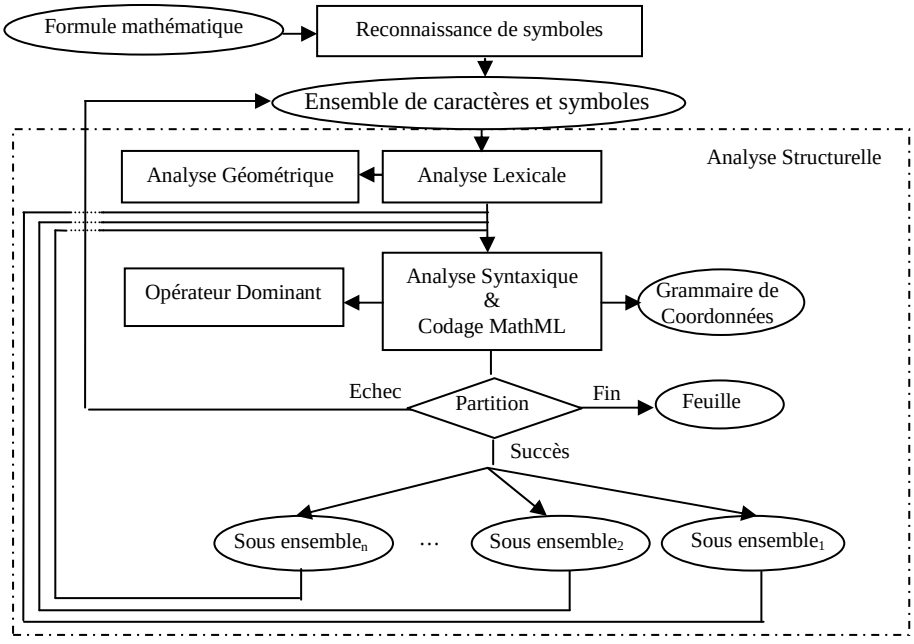


Figure 3. *Architecture du système proposé*

Les formules sont supposées êtres correctement extraites d'images de documents. Pour reconnaître les symboles, nous avons eu recours à la méthode des k-ppv et les moments de Hu. Pour l'analyse structurelle, nous avons procédé par une analyse lexicale, géométrique et syntaxique à la fois descendante et ascendante. Nous allons reprendre chacune de ces étapes, décrire son fonctionnement, les problèmes rencontrés ainsi que les propositions visant à leur résolution au moyen d'exemples de formules mathématiques.

4.1. *Reconnaissance des symboles mathématiques*

L'idée de base qui nous a servi de guide pour la reconnaissance des symboles mathématiques est la suivante : un symbole a de fortes chances d'être de la même « famille » ou classe que le ou les symboles qui lui ressemblent le plus, qui lui sont les plus «proches » dans l'ensemble des représentations. Il est donc nécessaire de modéliser cette notion de proximité ou de degré d'appartenance. Nous avons utilisé l'algorithme de k-ppv qui est parmi le plus simple de tous les algorithmes d'apprentissage. Ainsi, un symbole est classifié par un vote majoritaire de ses voisins, avec le symbole étant assigné à la classe la plus commune parmi ses k plus proches voisins. Les voisins sont pris d'un ensemble de symboles pour lesquels la classification correcte est connue. Cela peut paraître comme une base d'apprentissage pour la classification, bien qu'aucune étape d'apprentissage explicite ne soit exigée. Pour cerner les voisins, les symboles sont ici représentés par des moments de Hu dans un espace multidimensionnel. Les invariants algébriques sont obtenus à partir de quotients ou de puissances de moments. Un moment est une somme sur tous les pixels du modèle d'image pondérée par des polynômes liés aux positions des pixels. Ces moments sont invariants par translation et changement d'échelle. Nous avons calculé plusieurs distances (euclidienne, chebychev, canberra, etc.) pour décider au sujet des k plus proches voisins au symbole en question. Nous avons retenu la distance de Canberra qui offre le meilleur taux de reconnaissance.

Notre système est capable de reconnaitre 40 classes de symboles différents dont 3 sont des signes diacritiques ('.', '..', ',', etc.), 11 sont des opérateurs (, +, -, ←, *, , , , /, <, ×), 9 sont des lettres de l'alphabet (, , , , , , ,), 2 noms de fonction (,), 13 chiffres (, , , , , , , , 0, 1, 2, 5, 9) et 2 délimiteurs ('(' et ')'). Pour évaluer la performance de la reconnaissance des symboles, nous avons utilisé, pour chaque classe, une base de test de 20 échantillons. Nous avons abouti à un taux de reconnaissance de l'ordre de 70%. L'analyse des résultats obtenus a permis de cerner les cas de confusions. Nous avons remarqué que les symboles symétriques (ex. : '<' vs '>', '(' vs ')',…) ou à morphologie semblables (ex. : la barre de fraction horizontale vs le signe moins) sont généralement confondus. Cela s'explique par le fait que les moments de Hu, ici utilisés comme descripteurs des symboles, sont invariants par homothétie, par translation et changement d'échelle. Nous persistons à croire que l'ajout de caractéristiques sensibles à la symétrie devrait accroître le taux de reconnaissance et réduire les cas de confusion entre symboles. Nous avons distingué trois types de symboles symétriques : 1) les symboles symétriques par rapport à l'axe vertical et l'axe horizontal passant par leur milieu (ex. : =, ×, *, −, +), 2) les symboles symétriques par rapport à l'axe vertical

par leur milieu (ex. : ∏, ∩, ∪) et 3) les symboles symétriques par rapport à l'axe horizontal par leur milieu (ex. :>, <, ⊃, ⊂, (,), [,], {, }, ∑). Nous travaillons actuellement sur l'amélioration du taux de reconnaissance des symboles. Il importe de noter aussi que chaque symbole est représenté par une liste ordonnée de candidats potentiels, et si le résultat n'est pas satisfaisant le système s'intéressera au candidat suivant lors de l'analyse structurelle.

4.2. *Analyse lexicale*

Elle sert à typer l'ensemble des symboles reconnus. Ce typage permet une abstraction dans le cadre de l'analyse structurelle, en créant des classes de symboles ayant un comportement identique (voir tableau 1).

Type lexical	**Lexème**	**Description informelle**
SP	∑, ∏, جذ, جذ, جذ	Symboles de sommation ou de produit
SR	√	Symbole de racine
SI	∫	Symbole d'intégrale
BFH	—	Barre de fraction horizontale
GD	|, (,), {, }, [,]	Grands délimiteurs, crochets, accolades ouvrantes ou fermantes
PD	(,), {, }, [,], |	Petits délimiteurs, crochets, accolades ouvrantes ou fermantes
OP	=, ≠, <, >, ≥, ≤, +, *, ×, /, ,, ‘ ,←, ∩, ∪, ⊃, ⊇, ⊄, ⊂, ⊆, ∈, ∉	Opérateurs arithmétiques, de comparaison, ensemblistes, flèches, virgule orientée vers le haut ou le bas
Entier non signé	٣٥ ,١٢ 125, etc.	Les entiers non signés
Réel non signé	٣.٢, 78.5, etc.	Les réels non signés
Lettre	أ-ي	Les glyphes des lettres isolés
NF	حا, فتا, فا, لو, etc.	Abréviation de fonctions élémentaires usuelles
LM	نهـــا	Limite
Det	حد	Déterminant

Tableau 1. *Types lexicaux*

Les symboles multi-composés (ex. ±, ⊇, ⊆, ≥, ≤, etc.) nécessitent d'être rattachés par groupement vertical . Les abréviations multi-composées de fonctions doivent aussi être groupées mais horizontalement (ex. زجا, زجتا, زظا, زظتا, زقا, زقتا). A la suite de l'étape d'analyse lexicale, nous disposons pour chaque symbole des attributs associés tels que le type lexical, son emplacement, son étendue, sa position par rapport à la ligne de base, son identité. Le but des analyses suivantes est de déterminer la structure hiérarchique des symboles, représentée souvent par un arbre

de relations. Lors de l'analyse géométrique, nous avons défini dix relations spatiales que le symbole peut avoir avec son voisinage : H(en haut), B (en bas), D (à droite), G (à gauche), EG(en exposant à gauche), ED (en exposant à droite), IG(en indice à gauche), ID (en indice à droite), I (inclus) et L (limité par deux délimiteurs).

4.3. *Analyse syntaxique*

Dans cette phase, nous proposons une extension du formalisme grammaire de coordonnées, initialement adopté par [Anderson 77], dans le but de réduire les restrictions sur le positionnement de symboles et gagner en efficacité. Anderson a utilisé une approche descendante pour l'analyse des expressions mathématiques arithmétiques et matricielles les plus fréquemment utilisées. Pour améliorer cette analyse, nous proposons d'utiliser des connaissances sur les conventions de notation mathématique telles que la dominance et la priorité d'opérateurs pour éviter le retour arrière et gagner ainsi en efficacité. De plus, nous nous sommes inspirés des travaux de [Belaïd et Haton 84] en adoptant une analyse syntaxique à la fois descendante et ascendante. Il s'agit de diviser la formule en sous-formules plus simples par l'opérateur dominant qui à partir duquel et de ses contextes (à droite, à gauche, en haut, en bas, en exposant, en indice, dedans et délimité), l'analyseur choisit la règle correspondante dans la grammaire. Ce processus se répète d'une manière récursive jusqu'à ce qu'ou chaque but soit atteint ou bien toutes les possibilités échouent. Dans ce qui suit, nous expliquons comment choisir l'opérateur dominant.

4.3.1 *Opérateur Dominant*

L'opérateur dominant n'est pas nécessairement au début ou à la fin de la formule. Il peut être explicite, représenté par un symbole, ou implicite, indiqué par l'arrangement spatial de ses opérandes (ex. indice, exposant, multiplication implicite). Notre choix d'opérateur dominant se base sur la notion de dominance et de priorité particulièrement s'il y a des opérateurs qui ne sont pas alignés. Les étapes sont les suivantes :

- Calcul, pour chaque opérateur O, combien de fois il a été dominé par les autres opérateurs. Un opérateur O_1 domine un opérateur O_2 si O_2 se trouve dans l'étendue d'O_1. L'étendue d'un opérateur correspond aux emplacements possibles de ses opérandes.
- L'opérateur candidat à être dominant est celui qui est le moins dominé. Dans la figure 4, la barre de fraction horizontale n'a pas été dominée par aucun autre opérateur notamment la racine et l'exposant qui se trouvent dominés au moins une fois.

Figure 4. *Un seul opérateur dominant*

— S’il y a des opérateurs de même dominance, c’est le plus prioritaire qui l’emporterait selon l’ordre suivant : opérateurs de comparaison (=, ≥, ≤, ≠, …), les parenthèses, les opérateurs arithmétiques binaires (‘+’ et ‘-‘, ‘*’, ‘×’ et ‘/’), les opérateurs unaires ‘+’, ‘-‘ et ‘±’ et les indices et les exposants. Dans la figure 5, les opérateurs ‘×’ et ‘=’ ont la même dominance mais le ‘=’ est plus prioritaire, il constituera un candidat d’opérateur dominant.

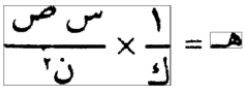


Figure 5. *Opérateurs de même dominance mais à priorité différente*

— S’il y a des opérateurs de même dominance et priorité, c’est le plus à droite qui sera choisi vu que les formules mathématiques Arabes s’écrivent de droite à gauche. Dans la figure 6, les deux signes d’addition ont la même dominance et priorité. Ainsi, le signe le plus à droite serait probablement l’opérateur dominant.

Figure 6. *Opérateurs de même dominance et priorité*

— Il faut vérifier aussi que l’opérateur dominant englobe, avec son étendue, la totalité de l’espace qu’occupe la formule. Dans le cas contraire, il s’agit d’une multiplication implicite entre deux sous formules (voir figure 7).

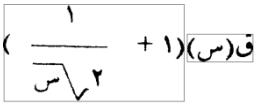


Figure 7. *Multiplication implicite de deux sous formules*

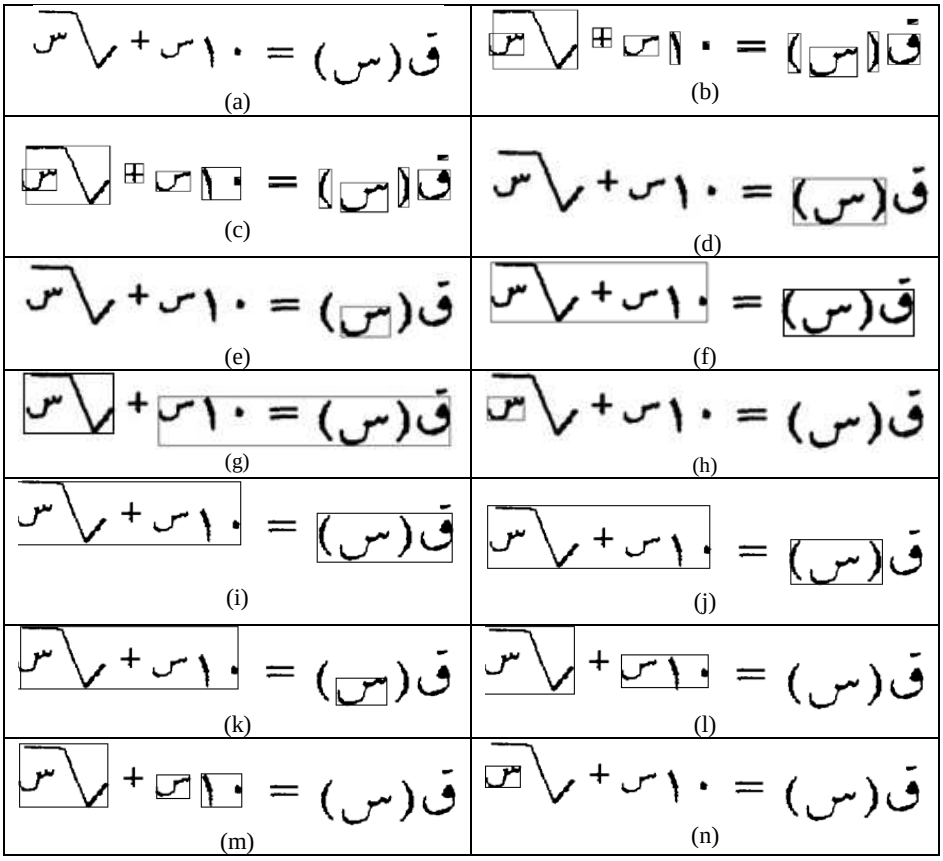
— S’il n’y a pas d’opérateurs dans la formule (voir figure 8), nous concluons une multiplication implicite entre variables ou constantes désignées par des lettres.

Figure 8. *Multiplication implicite entre variables ou constantes*

4.3.2 *Fonction d’Analyse*

Il s’agit de diviser récursivement la formule en sous formules par l’opérateur dominant. La stratégie de division est comme suit. Si l'ensemble de symboles ou de caractères, ne contient pas l'opérateur dominant ou bien le contexte n’est pas vérifié, la règle ne s'applique pas. S’il existe alors la division du reste des symboles ou des

caractères dans l'ensemble sera basée sur leur position relative à cette instance d'opérateur dominant. Nous avons établi une trentaine de règles syntaxiques. La figure 9 montre un exemple illustratif d'analyse d'une formule.



(a) Formule à analyser.
(b) Extraction des composantes connexes et reconnaissance des symboles : ق : lettre ق,) : parenthèse ouvrante, (: parenthèse fermante, س: lettre س, = :signe d'égalité, + : signe plus, ١ : chiffre 1, ٠ : chiffre 0 et √ : symbole de racine.
(c) Analyse lexicale : ق : NF,), (: PD, س: lettre, =, + : OP, ١٠ : entier_non_signé et √ : SR. La lettre « ق » a été considérée comme nom de fonction car elle se place juste au début de la formule et devant une parenthèse. Les chiffres ١ et ٠ sont groupés pour former un entier non signé.
(d) Analyse géométrique : calcul de l'étendue de ق
(e) Analyse géométrique : calcul de l'étendue des petits délimiteurs
(f) Analyse géométrique : calcul de l'étendue du signe d'égalité
(g) Analyse géométrique : calcul de l'étendue du signe plus
(h) Analyse géométrique : calcul de l'étendue du symbole de la racine

(i) Analyse syntaxique : Etape 1 : opérateur dominant : '='. La règle *E op E←E* est alors appliquée. Elle partitionne la formule en deux sous formules à gauche et à droite de l'opérateur ' =' et appelle la fonction d'analyse syntaxique pour les deux sous formules..
(j) Analyse syntaxique : Etape 2 : opérateur dominant de la sous formule de droite est le nom de fonction 'ق '. La règle $E\ ^{x}FN \leftarrow E$ est appliquée pour générer l'expression parenthesée.
(k) Analyse syntaxique : Etape 3 : opérateur dominant : les petits délimiteurs l'analyseur applique la règle *PD E PD← E* qui appelle l'analyseur syntaxique avec l'expression à l'intérieur des délimiteurs. Les règles *T←E* puis *V←T* puis *lettre←V* sont appliquées pour avoir la lettre ' س '.
(l) Analyse syntaxique : Etape 4 : opérateur dominant de la sous formule de gauche est le '+' et on applique la règle *E op E←E* qui génère les deux sous formules de droite et gauche.
(m) Analyse syntaxique : Etape 5 : opérateur dominant de la sous formule à droite du '+' est la multiplication implicite entre l'entier 10 et la lettre « س ». On applique la règle *E E ← E* pour diviser la sous formule. Ensuite, on applique *T←E* puis *entier_non_signé←T* pour retourner l'entier non signé١٠. On applique ensuite les règles *T←E* puis *V←T* puis *lettre←V* pour retourner la lettre ' س '.
(n) Analyse syntaxique : Etape 6 : opérateur dominant de la sous formule à gauche du signe '+' est le symbole de la racine. On applique la règle $E\ ^{R}SR \leftarrow E$. Elle retourne la sous formule à l'intérieur de la racine. On applique les règles *T←E* puis *V←T* puis *lettre←V* pour retourner la lettre ' س '.

Figure 9. *Exemple d'analyse*

A la fin de l'analyse structurelle, notre système génère le code MathML suivant :

```
<math dir="rtl">
<mrow>
<mroot><mn>س</mn></mroot><mo>+</mo><mn>س</mn><mo>&
invisibletimes;</mo><mn>10</mn><mo>=</mo><mfenced
open="(" close=")"><mn>س</mn></mfenced><&Applyfunction
;><mo>ق</mo>
</mrow>
</math>
```

Il est possible d'utiliser des informations structurelles pour aider à deviner les symboles ambigus ou en confusion. Pour ne pas confondre entre le signe moins et la barre de fraction horizontale, un contexte est défini dans la grammaire : pour qu'un symbole soit une barre de fraction, il devrait ne pas avoir d'étendues vides au-dessus et au-dessous de lui. Si le contexte n'est pas vérifié, il faut essayer avec les autres candidats issus de la reconnaissance des symboles.

5. Conclusion et perspectives

Cet article aborde la problématique de la reconnaissance hors ligne des formules mathématique Arabes typographiées. Pour reconnaître les symboles mathématiques, nous avons utilisé la méthode des K-ppv avec les moments de Hu. Pour l'analyse structurelle, notre choix d'une méthode syntaxique repose essentiellement sur la nécessité de définir une formalisation grammaticale des formules mathématiques Arabes ainsi que de déterminer les connaissances sur les conventions de notations

mathématiques utiles à une meilleure reconnaissance. Nous avons testé notre méthode sur 220 formules mathématiques Arabes et nous avons abouti à un taux fort satisfaisant (près de 90% des formules sont parfaitement reconnues). Nous nous attendons à une nette amélioration du taux de reconnaissance avec l'ajout de caractéristiques sensibles à la symétrie lors de la reconnaissance des symboles. Nous projetons dans l'avenir d'élargir le spectre des symboles utilisés et tenir compte de plus de variabilité au niveau de l'écriture des formules mathématiques Arabes.

6. Références

Khalid S., Sur certains aspects de la formulation et de l'écriture de la mathématique en langue arabe, Thèse de doctorat, Université Catholique de Louvain, Louvain-La-Neuve, 1992.

Lazrek A., « Aspects de la problématique de la confection d'une fonte pour les mathématiques arabes », *Cahiers GUTenberg* no 3940—Mai 2001.

R. H. Anderson, «Two-Dimensional Mathematical Notation», *in proc. of Syntactic Pattern Recognition Applications*, K.S. Fu, Ed. Springer Verlag, New York , pp. 147-177, 1977.

Belaïd A. and Haton J. P., «A syntactic approach for handwritten mathematical formula recognition», *IEEE Trans. Pattern Analysis and machine intelligence*, vol. 6, no. 1, pp. 105-111, 1984.

Blostein D. et Grbavec A., « Recognition of Mathematical Notation", *Handbook of character recognition and document image analysis* », *Eds. H. Bunke and P.S.P. wang, world scientific publishing company*, pp. 557-582, 1997.

Koschinski M., Winkler H. J. and Lang M., « Segmentation and recognition of symbols within handwritten mathematical expressions », *in Proc. of CASSP*, vol. 4, Detroit, MI, pp. 2439-2442, 1995.

Winkler H. J., Fahner H. and Lang M., « A soft decision approach for structural analysis of handwritten mathematical expressions », *in Proc. of ICASSP*, vol. 4, Detroit, MI, pp. 2459-2462, 1995.

Kosmala A., Rigoll G., Lavirotte S. and Pottier L., « On-line handwritten formula recognition using hidden Markov models and context dependent graph grammars », *in Proc. of ICDAR*, Bangalore, Karnataka, India, pp. 107-110, 1999.

Kacem A., Belaïd A. and M. Benahmed, « Automatic extraction of printed mathematical formulas using fuzzy logic and propagation of context », *in IJDAR : International Journal on Document Analysis and Recognition*, volume 4, Number 2, pp. 97-108, December 2001.

Topia E. and Rojas R., «Recognition of on-line handwritten mathematical formulas in the E-chalk system », *in Proc. of ICDAR*, Edinburgh, U.K., pp. 980-984, 2003.

Garain U., Chaudhuri B. and Chaudhuri A. Ray, « Identification of embedded mathematical expressions in scanned documents », *in Proc. of ICPR*, Cambridge, UK, pp. 138-149, 2004.