

Vers une Approche Fonctionnelle de la Résolution de la Référence dans le Dialogue Finalisé

Guillaume Pitel

LIMSI – CNRS

Bât. 508 BP 133 F-91403 Orsay CEDEX

Guillaume.Pitel@limsi.fr – Soutenance prévue en 2004

Keywords – Mots Clefs

Résolution de la Référence Intrinsèque, Approche Fonctionnelle, Dialogue Homme Machine
Reference Resolution, Intrinsic Reference, Functional Approach, Human Computer Dialogue

Résumé – Abstract

Dans cet article, nous montrons l'insuffisance du pouvoir d'expression des approches par prédicats pour la *résolution de la référence en extension* dans un cadre générique de dialogue homme-machine. Cette insuffisance oblige pour l'instant les concepteurs de tels systèmes de dialogue à concevoir des heuristiques *ad hoc* impossibles à intégrer dans un cadre de description unifié. Nous montrons que la résolution des expressions référentielles nécessite la prise en compte du contexte même pour les termes portant sur des caractéristiques intrinsèques aux éléments. Nous proposons alors un formalisme pour représenter la sémantique des **extracteurs référentiels intrinsèques**. Ce formalisme repose sur trois fonctions, la première permet de calculer le rapport de similarité de deux éléments en fonction d'une certaine dimension et dans un certain contexte, les deux autres permettent de partitionner un domaine de référence trié par l'utilisation de la première fonction.

In this paper, we show the lack of expressiveness of predicative approaches for extensional reference resolution. This drawback forces dialogue systems designers to develop specific heuristics for solving some reference cases they can't integrate in a unified representation framework. As we focus on extensional reference resolution in a generic framework for human computer dialogue system, we point the fact that the process of resolving referential expressions needs to take context into account even for expressions involving intrinsic characteristics of elements. Thus we propose a representation model for the semantic of **intrinsic referential extractors**. This model is built on three functions, the first one serves to calculate the similarity ratio between two elements, the two others serve to partition a reference domain which is previously sorted by the first function.

1 Introduction

La généricité est un objectif crucial dans le dialogue homme-machine, et ce pour deux raisons, l'une quantitative : le coût de développement et d'adaptation des systèmes de dialogue ; l'autre qualitative : la cohérence théorique du cadre dans lequel sont construits ces systèmes de dialogue. Nous étudions le cas *extensionnel* des *expressions référentielles*, défini par (Byron et Allen, 2002) comme la recherche des *éléments de l'application médiée*¹ désignés dans un énoncé². Ce traitement a longtemps été soit construit comme un filtrage successif par prédicats, soit construit autour d'heuristiques spécifiques (Winograd, 1973). Ces deux approches ne permettent pas de définir un cadre générique pour la résolution de la référence dans le cadre du dialogue finalisé. Dans cet article, nous présentons un modèle permettant de représenter les *extracteurs référentiels* de manière fine et qui s'appuie sur le modèle de résolution de la référence de (Salmon-Alt, 2001). Dans la terminologie de cet auteur, les termes composant une expression référentielle et que nous désignons par « *extracteurs référentiels* », sont représentés par des *critères de différenciation*, utilisés dans le processus de sélection d'élément³, et l'ensemble des entités à partir duquel l'opération de sélection s'opère est représenté par des *domaines de référence*. Nous adoptons sur les extracteurs référentiels un point de vue procédural, en les envisageant en tant qu'opérateurs actifs de sélection d'éléments dans un ensemble, plutôt que comme des formules de logique utilisées pour sélectionner les éléments répondant à certaines conditions de vérité. L'idée directrice est d'utiliser des **fonctions de différenciation** à la place des critères, nous montrons en effet dans la section 2 que représenter l'action des extracteurs référentiels par des prédicats ne permet pas de prendre en compte leur dépendance vis-à-vis du contexte. Nous proposons alors dans la section 3 de considérer l'acte d'extraction comme une différenciation dépendante du contexte, effectuée par ces fonctions. Dans la conclusion, section 4, nous présentons les problèmes que nous n'avons pas pu traiter jusqu'à maintenant, et proposons quelques pistes de recherche pour les aborder.

2 Etude d'extracteurs référentiels intrinsèques

Dans une expression référentielle qui désigne un ou des objets du monde, nous appelons *extracteurs référentiels* les termes jouant un rôle dans la construction de l'extension de l'expression par extraction d'éléments dans la représentation du monde. Le problème de la

¹ Nous désignons sous le terme « *application médiée* » le programme informatique avec lequel on souhaite que l'utilisateur puisse interagir par l'intermédiaire de la langue.

² Le traitement extensionnel s'oppose au traitement anaphorique, consistant à chercher un antécédent linguistique dans les énoncés passés, et au traitement logique consistant à construire une représentation logique d'une expression référentielle sans prendre en compte l'état de l'application médiée.

³ Les critères de différenciation sont classiquement utilisés par les modules de traitement de la référence pour représenter la sémantique des expressions référentielles. Chaque élément de l'expression est transformé en prédicat (par exemple « grand carré » → *grand(x)* ET *carré(x)*), et utilisé ensuite pour filtrer successivement les entités accessibles à l'interlocuteur afin de produire le ou les éléments compatibles avec l'expression référentielle.

représentation des objets⁴ de l'*application médiée* et de leur lien avec les expressions référentielles qui les désignent n'a pas été beaucoup exploré. La représentation sur laquelle se fait la recherche référentielle est généralement une base de connaissances composée d'objets adéquats pour l'application ou même représentée en logique du premier ordre. On constate ceci même dans les travaux récents, comme (Salmon-Alt, 2001 :149), (Byron et Allen, 2002) ou (Dale et Reiter, 1995). Typiquement, (Byron et Allen, 2002) expriment les champs de restriction par des propositions de la forme (color x RED). (Dale et Reiter, 1995) distinguent trois types de composantes pour caractériser un élément : les propriétés intrinsèques (p. ex. taille, couleur), les propriétés relationnelles (p. ex. position spatiale), les propriétés typologiques des objets (p. ex. triangle, carré). Nous étudierons dans cet article les extracteurs référentiels qui portent sur les caractéristiques intrinsèques. Les caractéristiques relationnelles pures, comme les prépositions spatiales (Dzikovska, 2000) ne seront pas étudiées ici.

Les extracteurs intrinsèques s'appuient sur les caractéristiques intrinsèques, qui sont des caractères stables (toujours présents) et indépendants du contexte. Nous allons nous attacher dans cette section à démontrer que même si ces caractères sont effectivement indépendants du contexte, il n'en reste pas moins que l'opération d'extraction du référent, effectuée dans les expressions qui font référence à ces caractères est, elle dépendante du contexte. Nous parlerons du **contexte opératoire** pour parler de l'état de l'application avec laquelle l'utilisateur dialogue ; du **contexte ontologique** pour parler de l'état des relations qui relient les différentes catégories d'entités entre elles ainsi que les connaissances typiques qui y sont liées ; du **contexte dialogique** lorsqu'il s'agira de faire référence à l'état de l'interaction entre l'utilisateur et l'application. Nous étudierons tout d'abord une caractéristique intrinsèque, la taille, dont il est communément admis que l'utilisation dans les expressions référentielles la transforme en une caractéristique relationnelle, et donc dépendante du contexte opératoire. Nous aborderons ensuite le cas de la couleur, généralement considérée comme intrinsèque et indépendante du contexte mais dont nous montrerons qu'elle peut aussi être dépendante du contexte.

2.1 La taille

Salmon-Alt étudie les variantes de la référence à partir d'un corpus de dialogues (relatant les échanges entre un utilisateur et un opérateur) destinés à faire réaliser un dessin complexe à l'opérateur à partir d'un ensemble d'objets décrits dans la Figure 1. Ces dialogues sont tirés d'un corpus construit à partir d'une expérimentation décrite dans (Ozkan, 1994). Dans cette situation, l'approche prédicative est tout à fait efficace, puisque toute entité X de l'application est représentée par un schéma de la forme suivante :

EntitéType X = (taille, couleur) où

- EntitéType \in {BARRE, TRIANGLE, ROND, CARRE}
- taille \in {GRAND, PETIT},
- couleur \in {BLANC, NOIR, ROUGE, VERT, BLEU}.

⁴ Nous distinguerons d'une part les représentations informatiques de l'application médiée, que nous appellerons **objets**, construits par un **type** et des **attributs** ; et d'autre part, les entités auxquelles l'utilisateur fait référence, que nous désignons par le terme **éléments**, ayant des **catégories** et des **caractéristiques**.

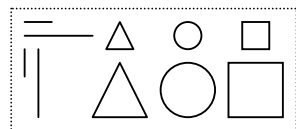


Figure 1 : Palette des formes disponibles, corpus Ozkan

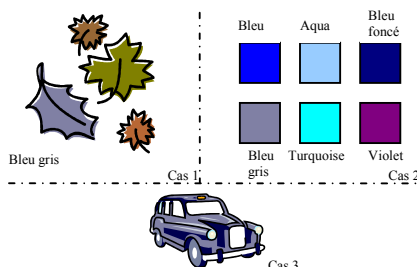


Figure 2 : Problème avec trois tailles (petit, moyen, grand) : le grand carré est-il en fait le carré moyen ?

Pourtant, il suffit d'ajouter une taille possible (MOYEN) pour que cette approche soit mise en défaut pour résoudre l'expression « *le grand carré* » dans le contexte présenté Figure 2. Le carré situé à gauche est en effet un grand carré par rapport à l'ensemble des carrés de la figure, mais son attribut *taille* ayant la valeur MOYEN, il ne peut pas être recruté comme candidat à l'expression référentielle « grand carré ». Comme le relève (Dale et Reiter, 1995) la taille définie dans {PETIT, MOYEN, GRAND} ne peut être liée aux adjectifs « petit », « moyen » et « grand », car ces termes désignent aussi la relation entre les tailles, que nous appelons *grandeur* pour la différencier. La taille contient uniquement les dimensions objectives des objets. Ceci nous amène à conclure que la sémantique *dénotationnelle* des adjectifs n'est pas, même dans un cas aussi simple que *grand* ou *petit*, réductible à une approche par prédicats de la logique du premier ordre.

2.2 La couleur

Nous considérons donc que la grandeur, contrairement à la taille, est une caractéristique dépendante du **contexte opératoire**. Le fait qu'il soit aisé de se méprendre sur ce point montre la difficulté qu'il peut y avoir à déterminer exactement la catégorie (typologique, intrinsèque ou relationnel) des caractères définissant les éléments.

Figure 3 : Catégorisation des couleurs en fonction de l'objet et du contexte⁵

Nous illustrons Figure 3 qu'il existe la même difficulté pour les adjectifs de couleur. En effet, dans le cas 1, « la feuille bleue » désigne la feuille de gauche (sur la version couleur), puis lorsque l'on demande à des sujets⁶ de désigner dans le cas 2 les carrés qui ne sont pas de couleur bleu, il choisissent les trois carrés du bas. Si l'on demande la couleur de la voiture dans le cas 3, la plupart des sujets répondent « noir », même s'ils répondent aussi pour la

⁵ Le sujet du présent article portant en partie sur la couleur, nous avons dû en utiliser dans nos schémas. Une version électronique est disponible à : <http://www.sciences.univ-nantes.fr/irin/taln2003>.

⁶ Nous avons procédé à une expérimentation avec une douzaine de sujets. Le protocole de test consiste en une feuille où est imprimée la figure avec les carrés colorés et des questions sur les objets.

plupart par l'affirmative si on leur demande si la voiture est bleue. Pourtant, la feuille, le carré en bas à gauche et la voiture ont tous la même couleur. Une feuille, qui est typiquement dans les couleurs vertes, jaunes, rouges ou marrons, peut donc être catégorisée comme bleue dès l'instant où sa couleur s'éloigne de ces couleurs typiques pour aller vers le bleu, tandis qu'une figure géométrique de la même couleur, qui en tant que figure a une couleur typique neutre, ne pourra pas être catégorisée comme bleue. C'est ce que nous appelons la *dépendance au contexte ontologique*. D'autre part, si un objet est perçu entouré par d'autres objets, la couleur de ces objets déterminera en partie la manière dont sa propre couleur sera catégorisée par un observateur humain. C'est ce que nous appelons la *dépendance au contexte opératoire*.

2.3 Importance du contexte pour certains attributs intrinsèques

D'après ces observations, le rôle du contexte ne peut pas être négligé si l'on s'intéresse au calcul référentiel des adjectifs de couleur. Cette conclusion rejoint celle de (Pateras *et al.*, 1995) portant sur des situations de dialogue où les caractéristiques utilisées dans la référence ne peuvent pas être mises en relation directe avec un schéma de base de donnée, ou plus généralement, avec une représentation prédicative des connaissances. La solution proposée pour choisir le référent correctement consiste à utiliser des fonctions floues d'appartenance à un ensemble. Cependant, ces fonctions sont définies par rapport à une situation spécifique, (voir Figure 4) et les auteurs ne proposent pas de moyen pour prendre en compte le contexte. Il en est de même dans (Lammens & Shapiro, 1993) où les auteurs construisent par apprentissage des fonctions de catégorisations de la couleur, fonctions qui sont au final indépendantes du contexte. Ces recherches montrent cependant clairement qu'une représentation sémantique des adjectifs nécessite un formalisme permettant d'exprimer des fonctions complexes, et permettant d'en définir précisément l'usage.

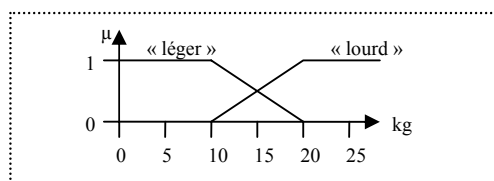


Figure 4 : Catégorisation du poids par une fonction floue d'après (Pateras, 1995)

Notre proposition est que la représentation sémantique des extracteurs référentiels intrinsèques devrait pouvoir supporter les cas d'utilisation suivants (ici avec l'exemple de la couleur bleue) : Choisir le bon référent en fonction du contexte. (« Supprime la feuille bleue » dans les Cas 1 et 2, Figure 3) ; Pouvoir dire si, dans un contexte donné, des objets peuvent être considérés comme bleus ou non (« Montre-moi les objets qui ne sont pas bleus » dans le Cas 2, Figure 3) ; Pouvoir être utilisé pour déterminer si X est plus ou moins bleu que Y (nous considérons que cette capacité permet aussi de traiter les superlatifs).

3 Modèle de différentiation fonctionnelle

Nous proposons de représenter les extracteurs référentiels, c'est-à-dire les termes de la langue qui servent à désigner un élément parmi d'autres, par des **structures de données contenant des fonctions**. Nous nous positionnons dans le cadre d'une application de dialogue où l'utilisateur peut parler d'éléments comme par exemple ceux visibles à la Figure 5. Nous

considérons pour l'instant que les éléments de l'application sous-jacente sont des objets (au sens algorithmique du terme) définis par leur type (ici tous sont de type *carré*) et par deux attributs : la couleur (un triplet de valeurs comprises dans $[0,1]$) et la taille (une valeur comprise dans $]0,+\infty[$).

3.1 Modèle d'extraction préliminaire

Nous proposons comme base préliminaire de réflexion de définir les extracteurs référentiels par une fonction de calcul de similarité qui suit. Nous discuterons ensuite cette première proposition et proposerons d'y adjoindre un mécanisme supplémentaire pour traiter les cas problématiques.

$$fSimil(entité\ a, entité\ b) \rightarrow [-1, 1] \cup \perp.$$

La fonction $fSimil$ doit permettre de trier un ensemble d'éléments en fonction d'une caractéristique (ici, bleu ou grand). Pour l'extracteur *bleu*, elle donne le rapport entre les couleurs de deux objets projetées selon la dimension bleue⁷. Si a est plus bleu que b , la valeur retournée est positive, négative dans le cas contraire, si l'une des deux n'a pas de position dans la dimension bleu, la valeur retournée est \perp . A partir de ces fonctions, il est alors possible de trier les entités du domaine de référence, et de séparer les entités à exclure des autres. Nous verrons par la suite que cette hypothèse ne permet pas de traiter certains cas de référence.

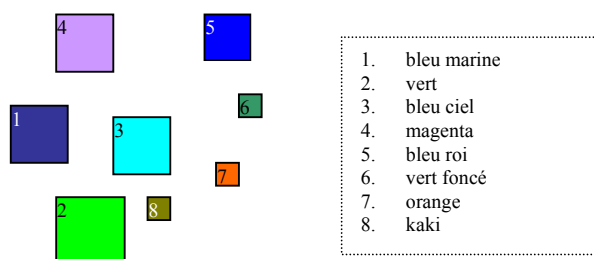


Figure 5 : Situation de test pour la représentation fonctionnelle des couleurs et des tailles.

A partir de la situation expérimentale exposée Figure 5, nous illustrons l'utilisation des fonctions de différentiation dans le cadre de la résolution de l'expression référentielle « *grand carré bleu* ». Nous prenons comme point de départ pour le domaine de référence l'ensemble des carrés de la scène (ici tous les objets). Nous appliquons la fonction de différentiation de l'extracteur référentiel « grand ». On suppose ici que la grandeur est une relation transitive, telle que $simil_G(x,z) = simil_G(x,y) * simil_G(y,z)$. On peut alors produire une liste triée d'entités (les numéros dans les cases sont les numéros des objets), étape 1 de la Figure 6, et annoter le lien entre deux cases par un rapport de similarité (ici case gauche par rapport à case droite) qui a été calculé entre elles. Puis nous appliquons, indépendamment de l'étape 1, la fonction de différentiation de l'extracteur référentiel « bleu ». En théorie, si la couleur est représentée

⁷ A ne pas confondre avec la composante bleue, dans le cas d'une décomposition en rouge, vert et bleu de la couleur. La *dimension bleue* est avant tout une dimension cognitive, qui doit être figurée à l'aide de fonctions mathématiques adéquates.

par un triplet de valeurs dans $[0,1]$, la fonction de différentiation appliquée à toutes les entités doit produire une matrice de $n \times n$ triplets. En pratique nous allons accepter ici une simplification consistant à calculer la distance entre le point bleu typique (qu'on suppose défini par défaut hors contexte, avant d'éventuellement le modifier) et le point de la couleur de l'entité. Dans ce cas, la distance entre deux couleurs sur la dimension bleue est simplement la différence des distances de ces deux couleurs par rapport à la couleur bleue typique. On peut alors, comme dans le cas de la grandeur, utiliser un algorithme de tri pour produire une liste triée d'entités, c'est l'étape 2 de la Figure 6. Il faut ensuite combiner les deux résultats. La fonction de combinaison de ces deux tris est la suivante : $simil(x,y) = \alpha \cdot simil_{Bleue}(x,y) \times \beta \cdot simil_{Grand}(x,y)$ où α et β sont des coefficients en rapport avec l'importance relative des caractères *couleur* et *taille* qui devraient être déduits à partir d'une expérimentation. Pour $\alpha = \beta = 1$, on obtient l'étape 3 de la Figure 6. Le résultat du calcul « grand carré bleu » appliqué à la situation Figure 5 donne donc le carré 1 comme référent préféré. La même méthode appliquée à l'expression « petit carré bleu » produit comme référent préféré le carré n°5. Dans les deux cas, cette méthode apporte un résultat cohérent avec ce à quoi on s'attend.

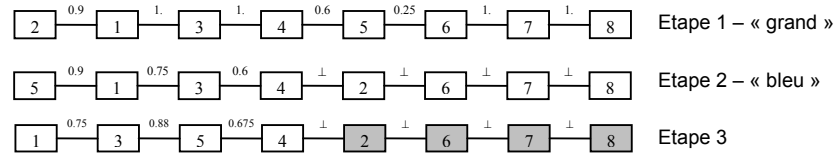


Figure 6 : Etapes de l'extraction de « grand carré bleu »

3.2 Modèle d'extraction étendu

3.2.1 Partitionnement

Il est nécessaire d'apporter des améliorations à la proposition préliminaire tout d'abord pour assurer la gestion de réponses négatives, autrement dit comment répondre : « Il n'y a pas de grand carré bleu », car, par exemple, la fonction de similarité de la grandeur ne permet pas l'**exclusion** de candidats (alors que la validation expérimentale sur l'énoncé « le grand carré orange » dans la situation décrite Figure 5 donne une réponse négative). D'autre part, le modèle de calcul référentiel doit être utilisable pour la sélection de groupes d'éléments, comme pour « les carrés bleus ». De même la modèle de calcul préliminaire ne permet pas d'expliquer le phénomène observé autour du cas 2 de la Figure 3 (dépendance contextuelle opératoire), qui montre que des couleurs acceptables pour un adjectif peuvent être exclues dans certaines situations et acceptées dans d'autres. La question est de savoir si le référent de l'expression « les carrés bleus » dans cette situation est l'ensemble des carrés $\{1,3,4,5\}$, $\{1,3,4\}$ ou $\{1,3\}$ ou encore un autre ? L'expérience nous montre que les sujets donnent majoritairement comme réponse l'ensemble $\{1,3,5\}$ ⁸. Ceci nous amène à conclure qu'il est nécessaire d'adjoindre un mécanisme de **partitionnement** similaire au mécanisme proposé

⁸ Dans le même ordre d'idées, il est à noter que la validation expérimentale sur « les petits carrés » produit la réponse $\{6, 7, 8\}$, alors que « le petit carré bleu » produit bien la réponse $\{5\}$, qui n'est pas dans l'ensemble $\{6,7,8\}$ bien que dans les deux cas, l'adjectif « petit » a été utilisé.

par Salmon-Alt, mais étendu pour obtenir des partitions ternaires. Pour cela, nous adjoignons à la fonction de similarité deux autres fonctions permettant de produire une partition ternaire sur le domaine de référence, découpée en **exclus**, **possibles** et **typiques**.

Concernant la dépendance au contexte opératoire, nous supposons qu'il n'influence pas la fonction de similarité. En revanche, la sélection des éléments exclus et typiques est, elle, dépendante de ce contexte, comme le montrent les phénomènes autour du cas 2 Figure 5. Les fonctions de partitionnement sont pour cette raison paramétrées par un domaine de référence, contenant tous les candidats potentiels à la référence. La dépendance des fonctions de partitionnement au contexte opératoire implique que les étapes d'extraction doivent être ordonnées. En effet, une étape a besoin du domaine de référence calculé par l'étape précédente pour partitionner correctement. Il est à noter que la phase de partitionnement négative (entités exclues) est obligatoire dans le processus, alors que la phase de partitionnement positive (entités typiques) n'est nécessaire que lorsque l'expression référentielle désigne un ensemble d'éléments.

Pour justifier l'ordonnancement des étapes d'extraction, nous nous appuyons sur les observations émises dans (Dale et Reiter, 1995) à propos de l'importance relative à attribuer aux caractères des objets pour la génération d'expressions référentielles. Ces observations ont une forte corrélation avec les maxims de (Grice, 1975), et suivent globalement le principe d'économie maximum. Les auteurs proposent de prendre en compte les caractères dans l'ordre suivant : type > propriété intrinsèque > propriété relationnelle. A partir de cette proposition, nous exprimons une règle plus générale pour l'ordonnancement par la proposition suivante : *Plus le contexte opératoire est significatif dans la sémantique d'un extracteur référentiel, plus tard cet extracteur référentiel doit être mis en œuvre dans la chaîne de résolution*. La perception des couleurs étant peu modifiée⁹ par le contexte opératoire, contrairement à la perception de la grandeur, l'ordre d'interprétation est donc bien *couleur* puis *grandeur*.

3.2.2 Représentation des extracteurs référentiels

D'après les propositions que nous avons émises précédemment, un extracteur référentiel est représenté par un agrégat de trois fonctions (Figure 7), et utilisé par l'algorithme Figure 8.

Une fonction *fSimil*, permet de trier des entités selon un critère donné en donnant un rapport de similarité entre deux entités selon ce critère. En plus des deux entités dont elle calcule le rapport, cette fonction peut prendre en argument un point de référence, optionnel ou non selon l'extracteur. Ce point de référence est déterminé par le contexte ontologique, c'est-à-dire par la catégorie des entités du domaine de référence sur lequel la différenciation est appliquée.

⁹ Pour comprendre le calcul du coût d'interprétation, il est nécessaire de voir le processus d'extraction référentielle comme un **algorithme paresseux**, ne prenant en compte les informations que lorsque c'est nécessaire. La prise en compte du contexte opératoire pour l'opération d'extraction sur la couleur n'est nécessaire que si le **rapport** entre les éléments d'un domaine de référence est **inférieur** à la variation maximale apportée par la prise en compte du contexte. Même si la couleur est tout autant dépendante du contexte que la grandeur, le poids du contexte est moins important sur la première que sur la seconde.

Une fonction *fExclus*, permettant de marquer le point d'exclusion dans le domaine de référence produit par le tri effectué avec la fonction *fSimil*. Pour l'extracteur *grand*, le point d'exclusion sera le dernier rapport de similarité entre entités strictement inférieur à 1 (c'est-à-dire les objets les plus petits). Pour l'extracteur *couleur*, ce point correspond au premier rapport de similarité égal à \perp .

Une fonction *fTypique*, permettant de marquer le point à partir duquel les entités ne sont plus typiques pour un extracteur donné. Nous n'avons pu déterminer de règle générale pour cette fonction, mais pour la taille par exemple, elle pourrait marquer le point où le rapport de similarité entre une entité et la première devient inférieur à un certain seuil, ou bien encore le point où le rapport de similarité d'entités consécutives tombe en dessous d'un certain seuil.

Extracteur :

- $fSimil(entité\ a, entité\ b, ref) \rightarrow [-1,1] \cup \perp$
- $fExclus(DomaineRef\ d) \rightarrow DomaineRef$
- $fTypique(DomaineRef\ d) \rightarrow DomaineRef$

Figure 7 : Représentation d'un extracteur référentiel avec prise en compte des marques d'exclusions et de typicalité.

1. Choisir le domaine de référence courant en fonction du type de l'expression référentielle (définie, indéfinie, etc.) tel que proposé dans (Salmon-Alt, 2001)
2. Parmi tous les extracteurs, faire dans l'ordre croissant de dépendance au contexte
 - a. Trier les entités non exclues du domaine de référence courant avec la fonction de similarité de l'extracteur référentiel.
 - b. Marquer la borne d'exclusion dans la liste triée.
3. Si la totalité du domaine de référence est exclu, fin et échec de la résolution
4. Si le type de l'expression référentielle nécessite de sélectionner les référents typiques, alors 5 sinon 6
5. Parmi tous les extracteurs, faire dans le même ordre que 2 : Marquer la borne typique dans la liste triée des candidats référents. (Le calcul de la borne typique est effectué en fonction des entités présentes dans le domaine non exclu)
6. A partir de la liste triée et marquée, extraire le ou les candidats référents vainqueurs en fonction du type de l'expression référentielle.
7. Mettre à jour le domaine de référence courant.

Figure 8 : Algorithme d'utilisation des extracteurs référentiels

4 Travaux en cours et perspectives de recherche

Après avoir constaté l'insuffisant pouvoir d'expression des approches par prédicats pour la résolution de la référence dans un cadre générique de dialogue homme machine, et que cette insuffisance oblige pour l'instant les concepteurs de tels systèmes à concevoir des heuristiques ad hoc impossibles à intégrer dans un cadre de description unifié, nous avons proposé un formalisme pour représenter la sémantique des extracteurs référentiels intrinsèques. La prise en compte des extracteurs relationnels et typologique nécessite d'autres mécanismes pour accompagner l'approche fonctionnelle que nous avons proposée, et nous préparons actuellement une publication à ce sujet. L'utilisation des fonctions de différentiation nous semble une voie prometteuse pour résoudre ces problèmes. Grâce à la représentation utilisée, il est très facile de prendre en compte les modificateurs d'intensité (très, peu, etc.) et la négation. Nous nous engageons dans une exploration approfondie de cette voie de recherche.

Actuellement, l'équipe AMI du LIMSI-CNRS développe le projet INTERVIEWS (Sansonnnet et al., 2002) (Sabouret et Sansonnnet, 2001) autour du concept d'agents conversationnels

d'assistance aux *gens ordinaires* pour l'aide à l'utilisation des composants et services WEB. Le formalisme présenté dans cet article est en cours d'implémentation dans ce cadre applicatif. Ce projet intègre aussi des mécanismes pour le traitement des expressions référentielles relationnelles et typologiques, issus de nos travaux, mais que nous n'avons pas pu présenter ici. Il nous est pour l'instant impossible d'en donner une évaluation, les principes sous-tendant son fonctionnement étant assez éloigné de ceux des outils existants, il nous est nécessaire de définir un grand nombre de connaissances avant de pouvoir l'exploiter.

Références

Byron D.K, Allen J.F. (2002) *What's a Reference Resolution Module to do? Redefining the Role of Reference in Language Understanding Systems*, Actes de DAARC2002.

Dale R., Reiter E. (1995), *Computational Interpretation of the Gricean Maxims in the Generation of Referring Expressions*. Cognitive Science.

Dsikovska M.O., Byron D.K. (2000), *When is a union really an intersection? Problems interpreting reference to locations in a dialogue system*, Actes de GOTALOG'2000.

Grice H.P. (1975). *Logic and conversation*. In P. Cole and J. Morgan, editors, *Syntax and Semantics: Vol 3, Speech Acts*, pages 43--58. New York, Academic Press.

Lammens J.M., Shapiro S.C., *Learning Symbolic names for Perceived Colors*, in Machine Learning in Computer Vision: What, Why and How?, AAAI-TR FSS93-04.

Ozkan N. (1994), *Vers un modèle dynamique du dialogue : analyse de dialogues finalisés dans une perspective communicationnelle*, Thèse, INP Grenoble.

Pateras C., Dudek G., Mori R. D. (1995), *Understanding Referring Expressions in a Person-Machine Spoken Dialogue*, Actes de ICASSP'95, Detroit, MI.

Sabouret N., Sansonnet J.P. (2001), *Un modèle de requêtes sur le fonctionnement de composants actifs*, Actes de MFI'01.

Sansonnet J.P., Sabouret N., Pitel G. (2002), *An Agent Design and Query Language dedicated to Natural Language Interaction*, Poster à AAMAS 2002.

Salmon-Alt S. (2001), *Référence et dialogue finalisé : de la linguistique à un modèle opérationnel*, Thèse d'informatique, Université H.Poincaré - Nancy 1, France. Mai 2001.

Schang D. (1995), *Application de la notion de cadre aux énoncés de positionnement et de référence*, Rapport de recherche n° 2529, Unité de Recherche INRIA Lorraine.

Winograd T. (1973), *A Procedural Model of Language Understanding*, Computer Models of Thought and Language, Roger Schank & Kenneth Colby éditeurs., W. H. Freeman Press.