Description des déterminants numéraux anglais par automates et transducteurs finis

Agata CHROBOT

Abstract: The cardinal and the ordinal numerals are closed and clearly delimited subsets of compound determinants, built upon the set of about seventy simple numerals (*zero, one, two, first, second,* etc.) and the *and* conjunction, so they can easily be described by finite-state machines.

INTEX, which is a large corpus processing system, allows to design different kinds of finite-state automata and transducers (represented as *graphs*), and to apply them in different ways in the process of lexical analysis of texts. This study presents two graph libraries for the recognition of English cardinal and ordinal numerals. The first, simpler and easily extensible, library contains "pure" finite-state automata (*i.e.* without output symbols). The second library, which consists of finite-state transducers, calculates the numerical value of a letter numeral (e.g. *eleven hundred → 1100*). It allows to formally express some ambiguities (e.g. *one billion* meaning 10^9 in American English and 10^{12} in British English) and equivalences (e.g. *one thousand and two hundred* meaning the same as *twelve hundred*) between different numerals.

Keywords: computational linguistics, finite-state automata and transducers, compound determinants, INTEX system.

Mots clés : traitement automatique du langage naturel, déterminants composés, automates et transducteurs finis, système INTEX.

1. Introduction

Le recensement et la description des mots composés constituent l'une des stratégies principales du projet linguistique du Laboratoire d'Automatique Documentaire et Linguistique (LADL), mené entre autres pour le français et l'anglais. Dans le système INTEX, créé en tant que

✉ Agata CHROBOT, LADL (Laboratoire d'Automatique Documentaire et Linguistique)
Université de Marne-la-Vallée, Institut Gaspard-Monge,
5, Bd Descartes, Champs-sur-Marne, F-77454 Marne-la-Vallée Cedex 2
Fax : ++33/144276851 e-mail : xsavary@free.fr

cadre informatique de ce projet, la description des mots composés peut être effectuée de deux façons (SILBERZTEIN 1990) :

- par une ou plusieurs listes textuelles appelées dictionnaires DELAC[1], où une ligne représente un mot composé sous sa forme de base, par exemple :

(1) *blue collar worker(worker.N1 :s),N+ANN:s/+N*[2]

- par automates finis et expressions rationnelles, ou par transducteurs finis, où chaque chemin entre l'état initial et l'état final représente un mot, simple ou composé. Ces outils de description sont les plus adaptés aux mots qui ont un nombre important de variantes, qui ont un degré relativement élevé de productivité, ou bien qui utilisent un petit nombre de mots simples pour créer un grand nombre de combinaisons. L'analyse combinatoire exhaustive de toutes les unités simples appartenant aux séquences à décrire est dans ces cas beaucoup plus facile à réaliser et à maintenir par des outils à états finis que par des listes textuelles.

Des familles importantes de composés ont déjà été décrites par des automates finis, par exemples les dates du français par Maurel (1989), ou les termes anglais de la bourse par Gross (1997). Dans cet article, nous présentons l'exemple des déterminants numéraux cardinaux et ordinaux anglais, où l'intérêt d'emplois des outils à états finis est évident : les numéraux constituent un ensemble a priori infini construit à partir d'une classe fermée de seulement quelques dizaines de mots simples (la conjonction *and* et 67 numéraux simples : *zero, one, two, ..., ten, eleven, ..., twenty, thirty, forty, ..., ninety, hundred, thousand, million, billion, trillion, quadrillion, first, second, third, ..., tenth, ..., billionth, trillionth, quadrillionth*[3]). Pour cette raison, et

---

[1] Dictionnaire Electronique du LADL des mots Composés.

[2] Ceci est un nom composé au singulier du type *Adjectif Nom Nom*, son constituant caractéristique, (*i.e.* le composant qui détermine les traits flexionnels du composé entier, cf. SILBERZTEIN 1990, et CHROBOT 1999) est le nom *worker* qui est au singulier et qui se fléchit selon le modèle (dit *code flexionnel*) *N1*. Ce nom composé admet la flexion en nombre (+*N*) qui est obtenue par la flexion du constituant caractéristique.

[3] Nous avons arrêté notre description à 10^{18}-1 car les nombres plus grands sont d'habitude exprimés par des puissances de 10.

aussi grâce au fait que les numéraux correspondent à des séquences contiguës dans des textes, ils peuvent être décrits indépendamment du reste des déterminants.

Nous décrivons les numéraux d'une part par des automates finis proprement dits (*i.e.* sans alphabet de sortie), et d'autre part par des transducteurs finis. Cette double description est motivée par deux méthodes différentes, disponibles dans le système Intex, d'application des graphes de composés à l'analyse lexicale d'un texte. La bibliothèque d'automates est plus simple et ainsi plus facilement extensible à des sous-ensembles plus larges de déterminants composés. La bibliothèque de transducteurs permet d'attacher aux séquences reconnues (en toutes lettres) leurs valeurs en tous chiffres, et ainsi d'exprimer certaines ambiguïtés et équivalences sémantiques présentes entre différents numéraux.

2. Déterminants numéraux cardinaux

Les déterminants numéraux peuvent s'écrire en tous chiffres (*25*), en toutes lettres (*twenty-five*), ou encore comme une combinaison de deux possibilités (*25.6 million*).

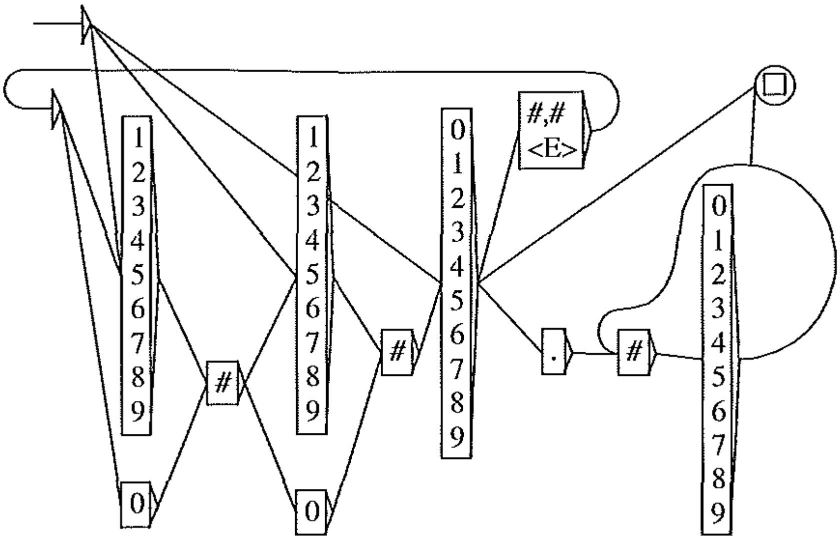

Figure 1. Graphe DnumDig *des numéraux cardinaux en tous chiffres*

Dans les nombres en tous chiffres (Fig. 1), les fractions sont séparées des entiers par un point, et les groupes de trois chiffres avant le point peuvent être séparés soit par un blanc (*251 554.05*), soit par une virgule (*251,554.05*).

Les numéraux en toutes lettres demandent un ensemble beaucoup plus élevé de graphes. Pour les cardinaux de 0 à 10^{18}-1 nous en avons construit 11. Nous utilisons le mécanisme d'imbrication des graphes (boîtes grisées) qui donne aux automates finis une puissance descriptive équivalente à celle des RTN (*recursive transition networks*). Néanmoins, grâce au fait que nos imbrications ne sont jamais récursives, le langage que nous décrivons est régulier [1].

Les numéraux simples entre 1 et 20 sont listés dans les graphes *DnumTxt1_9* (Fig. 2) et *DnumTxt10_19* (Fig. 3). Ce premier est imbriqué dans le graphe *DnumTxt20_99* (Fig. 4) pour les numéraux simples et composés de 20 à 99. Les trois graphes font partie du graphe *DnumTxt1_99* (Fig. 5) des numéraux entre 1 et 99. Celui-ci est ensuite inclus dans celui des cardinaux inférieurs à 1 000 (Fig. 6), et ainsi de suite.

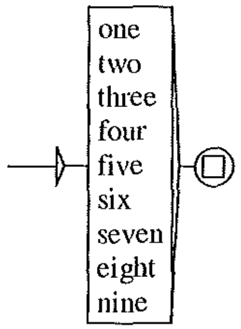

Figure 2. Graphe DnumTxt1_9 *: cardinaux de 1 à 9*

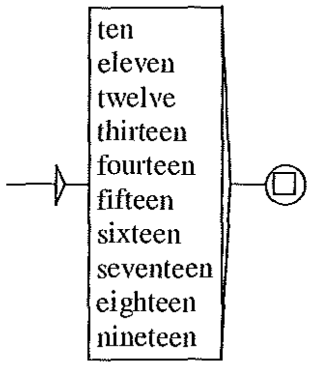

Figure 3. Graphe DnumTxt10_19 *: cardinaux de 10 à 19*

[1] En effet, considérons un graphe où toutes les imbrications sont non récursives. Nous pouvons remplacer chacun de ses nœuds grisés par le graphe qu'il représente. Par un nombre fini de telles substitutions nous pouvons obtenir un graphe sans nœuds grisés (et donc équivalent à un automate fini) qui reconnaît le même langage que le graphe de départ.

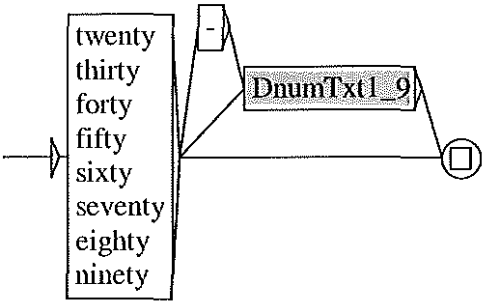

Figure 4. Graphe DnumTxt20_99 *: cardinaux de 20 à 99*

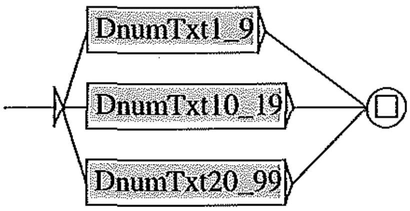

Figure 5. Graphe DnumTxt1_99 *: cardinaux de 1 à 99*

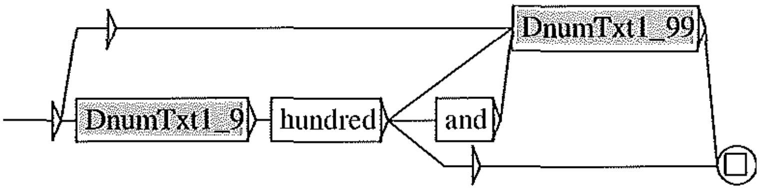

Figure 6. Graphe DnumTxt1_999 *: cardinaux de 1 à 999*

Analysons plus en détail le graphe *DnumTxt10^3_10^6-1* (Fig. 7) qui décrit en particulier les déterminants à double lecture tels que :

(2) *two thousand seven hundred and ten = twenty-seven hundred and ten (= 2710)*

La première lecture ci-dessus est réalisée par la deuxième branche du graphe et la deuxième par l'une des branches inférieures. Remarquons aussi que le nombre des milliers et des centaines qui les suivent peut être exprimé par un nombre en tous chiffres possédant éventuellement une partie non entière comme :

(3) *7.6 thousand (= seven thousand six hundred)*

ce qui est exprimé par le chemin supérieur du graphe.

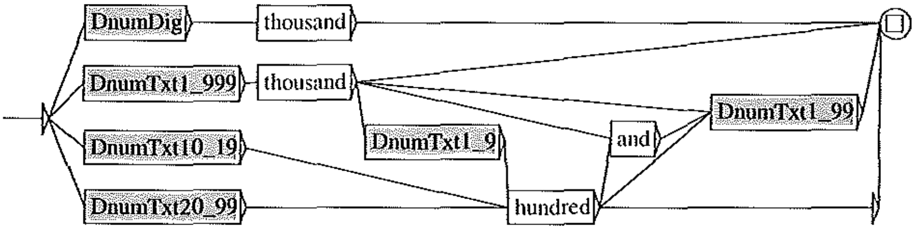

Figure 7. Graphe DnumTxt10^3_10^6-1 *: cardinaux entre 1 000 et 999 999.*

Le graphe des cardinaux entre 1 et 10^6-1 (Fig. 8) est obtenu par la concaténation de ceux des nombres inférieurs (Fig. 6) et égaux/supérieurs (Fig. 7) à 1 000.

Pour les nombres plus grands, le schéma des graphes est répétitif, avec trois branches dont une contenant des chiffres (*6.8 million*), la deuxième toujours contenant *DnumTxt1_10^6-1* suivi de *million*, *trillion*, etc., et la troisième le graphe construit à l'étape précédente. Ainsi, *DnumTxt1_10^6-1* (Fig. 8) est inclus dans *DnumTxt1_10^12-1* (Fig. 9), ce dernier, à son tour, dans *DnumTxt1_10^18-1* (Fig. 10), etc.

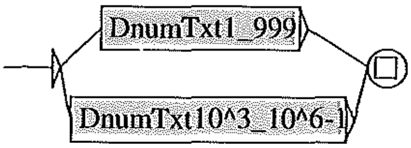

Figure 8. Graphe DnumTxt1_10^6-1 *: cardinaux positifs inférieurs à* 10^6.

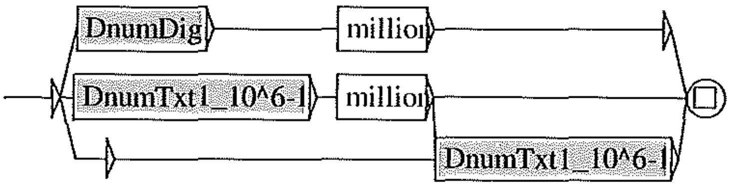

Figure 9. Graphe DnumTxt1_10^12-1 *: cardinaux positifs inférieurs à* 10^{12}

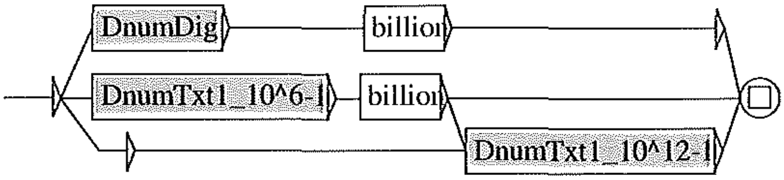

Figure 10. Graphe DnumTxt1_10^18-1 *: cardinaux positifs inférieurs à* 10^{18}

Le graphe principal *DnumTxt* (Fig. 11) contient le dernier graphe créé (ici nous nous sommes arrêtée à 10^{18}-1) ainsi que le déterminant *zero* qui n'a pas été pris en compte dans d'autres graphes car il correspond au chiffre *0* omis dans la lecture des cardinaux positifs.

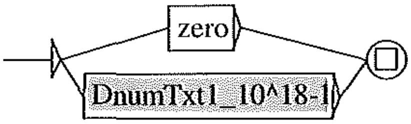

Figure 11.
Graphe DnumTxt *des déterminants numéraux cardinaux inférieurs à* 10^{18}.

3. La description des cardinaux par transducteurs finis

Il est souhaitable que l'analyse des déterminants par graphes permette d'attribuer aux séquences reconnues des étiquettes grammaticales contenant le lemme, la catégorie, les marques distributionnelles et les traits flexionnels, de la même façon que ceci a lieu pour les mots contenus dans les dictionnaires DELAF et DELACF (cf. SILBERZTEIN 1997). Ici, nous allons introduire des lemmes particuliers — chaque numéral reconnu sera accompagné de son équivalent écrit en chiffres, par exemple :

(4) *two thousand one hundred and ten,2110.DET+Num:p*

Nous pouvons obtenir ce type d'étiquettes par l'introduction de symboles de sortie (productions), marqués en dessous des nœuds des graphes. Lors du passage par l'un des chemins d'un graphe, les symboles de sortie sont concaténés et l'étiquette ainsi obtenue est rattachée à la séquence reconnue. L'ajout des telles productions augmente la complexité de la description pour trois raisons [1] :

1) les numéraux simples contenus auparavant dans le même nœud d'un graphe doivent être séparés à cause de leurs productions individuelles (par exemple Fig. 12 en comparaison avec Fig. 4) ;
2) la lecture des grands nombres (à partir de 10^9) n'est pas la même en anglais britannique qu'en anglais américain, ce que nous décrivons dans la suite ;
3) compte tenu du fait que le chiffre *0* est omis dans la lecture des cardinaux positifs (*201 = two hundred *zero one*), il faut introduire des ε-transitions avec sortie « 0 » pour permettre l'insertion de ce symbole à des positions non initiales. Ainsi, la création progressive

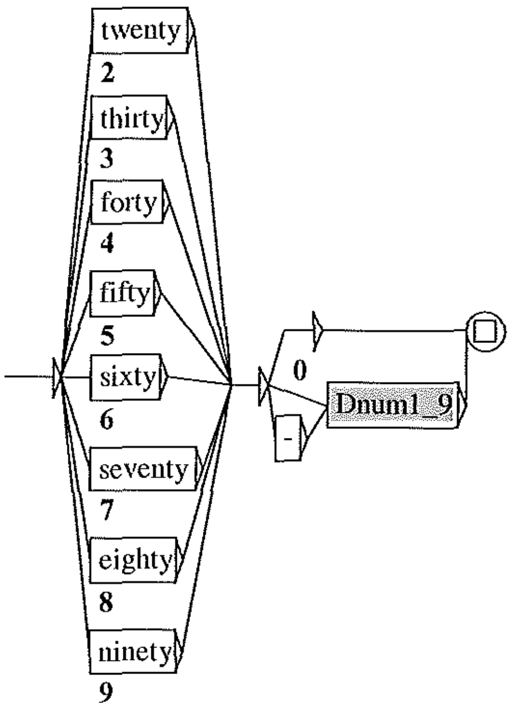

Figure 12. Graphe Dnum20_99 *: cardinaux de 20 à 99*

[1] L'augmentation de la complexité de description par transducteurs par rapport à celle par automates finis se reflète dans le nombre de graphes créés dans les deux cas. Pour les cardinaux inférieurs à 10^{18} nous avons obtenu 11 automates, mais les mêmes cardinaux représentés par transducteurs ont nécessité 23 graphes.

des graphes décrivant les numéraux de plus en plus grands ne suit pas les mêmes règles que dans le cas des automates finis (section précédente).

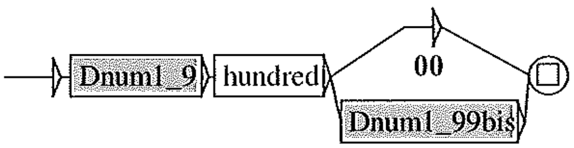

Figure 13. Graphe Dnum100_999 *: cardinaux de 100 à 999.*

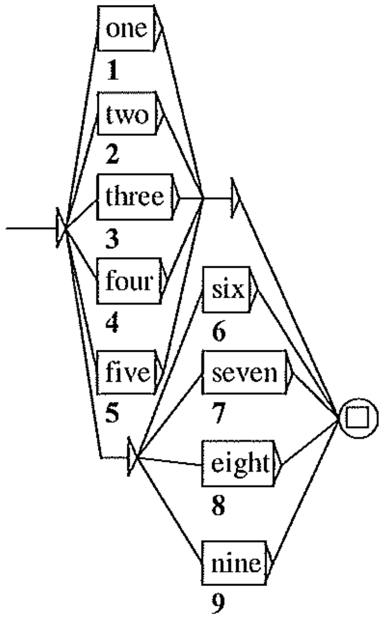

Figure 14. Graphe Dnum1_9 *: cardinaux de 1 à 9.*

Pour expliquer ce troisième phénomène, prenons l'exemple du graphe *Dnum100_999* (Fig. 13) qui décrit les cardinaux de 100 à 999. Pour exprimer le nombre des centaines, nous y imbriquons le graphe *Dnum1_9* (Fig. 14) créé auparavant. Mais pour décrire les dizaines et les unités nous ne pouvons pas nous servir des graphes déjà existant comme *Dnum20_99* (Fig. 12), car nous devons admettre les dizaines et/ou des unités nulles, *i.e.* des insertions du « 0 ». Le cas où à la fois les dizaines et les unités sont nulles est reconnu par l'ε-transition avec sortie « 00 ». De plus, nous créons un graphe auxiliaire *Dnum1_99bis* (Fig. 15) décrivant les nombres entre 1 et 99 qui sont précédés des centaines. La branche supérieure de ce graphe nous permet d'insérer le chiffre « 0 » à la positions des dizaines.

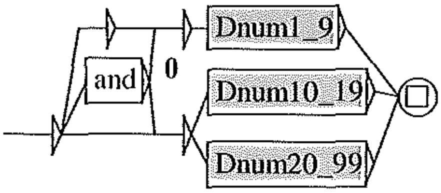

Figure 15. Graphe auxiliaire Dnum1_99bis *: cardinaux entre 1 et 99.*

Nous avons mentionné que la lecture des nombres en anglais britannique (AB) est différente de celle en anglais américain (AA) à partir de 10^9. D'après OALDCE (1989), on passe en AB d'un million à un billion, puis à un trillion, à un quadrillion etc. par ajout de 6 zéros, donc par multiplication par un million. En AA, ce passage à lieu déjà après l'ajout de 3 zéros, donc après la multiplication par un mille (voir Tab. 1).

Nombre	Anglais britannique	Anglais américain
1 000 000 000	one thousand million	one billion
1 000 000 000 000	one billion	one trillion
1 000 000 000 000 000	one thousand billion	one quadrillion
1 000 000 000 000 000 000	one trillion	one quintillion

Tableau 1.
Lecture de grands nombres en anglais britannique et en anglais américain.

C'est pourquoi le graphe principal *Dnum* (Fig. 16) des cardinaux inférieurs à 10^{18} contient trois parties. La première, qui représente les nombres inférieurs à 10^9 est commune pour l'AB et l'AA. Nous y distinguons les déterminants *zero* et *one* car ce premier est omis dans la lecture des cardinaux positifs et ce deuxième, reconnu seul, doit obtenir le trait morphologique *s* du singulier.

La deuxième partie du graphe *Dnum* correspond à la lecture britannique des nombres supérieurs ou égaux à 10^9. Suite au passage par cette branche, une séquence reconnue obtient une étiquette contenant le trait du dialecte *+GB*. La troisième partie reconnaît les versions américaines des mêmes cardinaux, et leur attribue le trait *+US*.

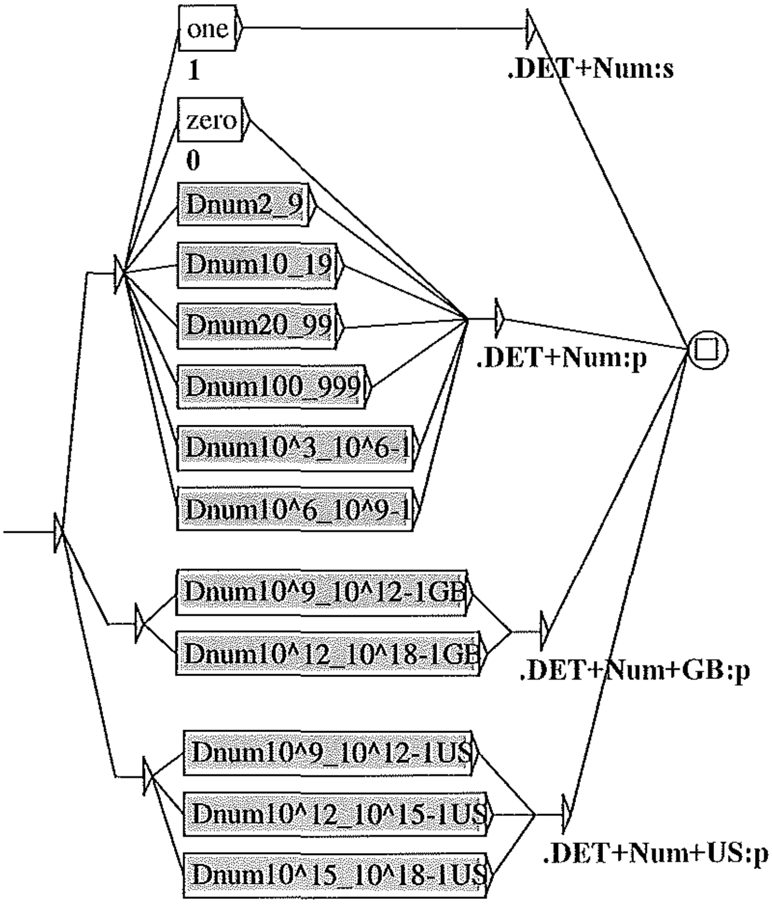

Figure 16. Graphe Dnum *: cardinaux inférieurs à 10^{18}.*

4. Déterminants numéraux ordinaux

La description des numéraux ordinaux, aussi bien par automates finis que par transducteurs, est parallèle à celle des cardinaux, mais il est nécessaire de traiter spécialement le dernier constituant qui porte la marque *st*, *nd*, *rd*, *d* ou *th*. C'est pourquoi le graphe *DnumOrdDig* (Fig. 17) des ordinaux en tous chiffres est considérablement plus complexe que celui des cardinaux (Fig. 1), même s'il ne contient pas la partie après le point correspondant à des fractions.

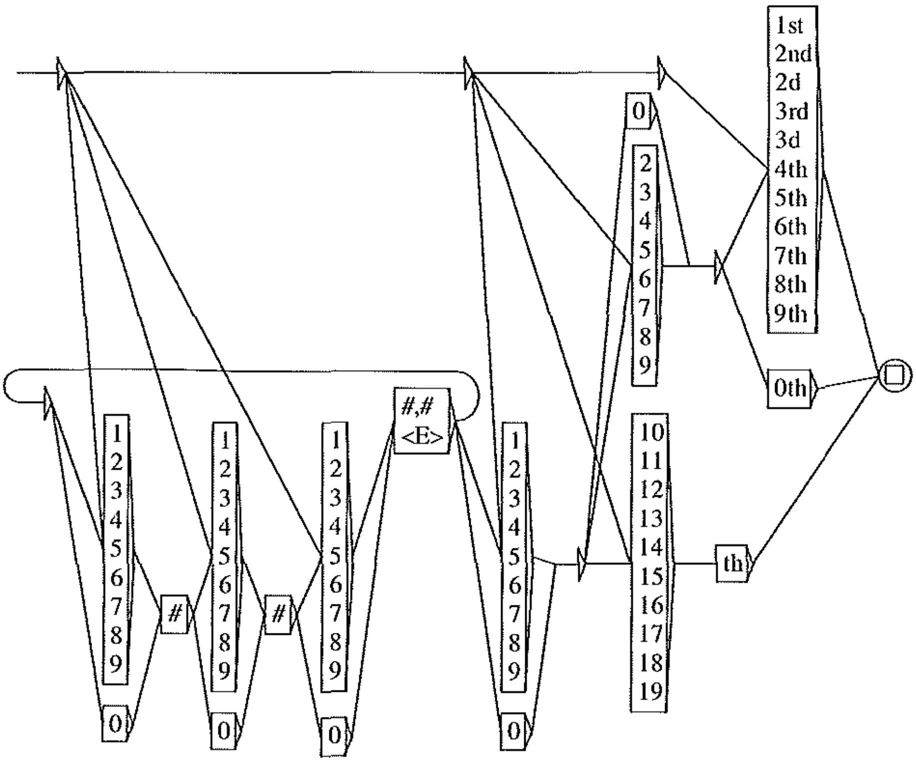

Figure 17. Graphe DnumOrdDig *des numéraux ordinaux en tous chiffres.*

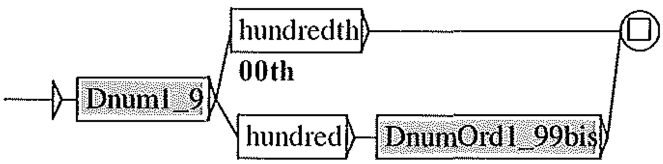

Figure 18. Graphe DnumOrd100_999 *: ordinaux de 100 à 999.*

Analysons aussi un exemple de graphe pour les ordinaux en toutes lettres. Le transducteur *DnumOrd100_999* (Fig. 18) des ordinaux entre 100 et 999 est obtenu de celui pour les cardinaux (Fig. 13) par ajout du symbole d'entrée « *hundredth* » au cas où les dizaines et les unités sont nulles (*seven hundredth*). Dans ce graphe, le nombre des centaines est exprimé par un numéral cardinal (*DnumTxt1_9*, Fig. 2), tandis que le nombre des dizaines et des unités est un numéral ordinal (*DnumOrdTxt1_99bis*). Voici des exemples de séquences reconnaissables par le graphe ci-dessus, avec les lemmes attachés au cours de l'analyse lexicale :

(5) *seven hundredth,700th.DET+Num+Ord*
(6) *seven hundred and first,701st.DET+Num+Ord*

5. Emplois des étiquettes grammaticales des déterminants numériques

Le rattachement d'une forme canonique en chiffres, qui n'est pas ambiguë, permet de tenir compte des ambiguïtés des numéraux écrits en toutes lettres. Par exemple le nombre suivant reçoit, à raison, deux étiquettes différentes avec la marque du dialecte (*US* pour l'anglais américain, *GB* pour l'anglais britannique) :

(7) *one billion,1000000000.DET+Num+US:p*
(8) *one billion,1000000000000.DET+Num+GB:p*

D'autre part nous pouvons rendre compte des équivalences entre des numéraux écrits différemment. Les deux premiers exemples ci-dessous, ainsi que les deux derniers, représentent les mêmes nombres, ce qui est explicité par leurs formes canoniques :

(9) *one thousand nine hundred seventy-four,1974.DET+Num:p*
(10) *nineteen hundred seventy-four,1974.DET+Num:p*
(11) *one billion,1000000000.DET+Num+US:p*
(12) *one thousand million,1000000000.DET+Num+GB:p*

6. Extension de la grammaire

Les déterminants numéraux constituent un sous-ensemble fermé et bien délimité de tous les déterminants composés. Grâce au mécanisme d'imbrication de graphes, nous pouvons élaborer une grammaire plus générale de déterminants composés contenant des numéraux. Dans ce cas, la première bibliothèque de graphes, *i.e.* les automates sans sorties, doivent être utilisés. Par exemple dans la phrase suivante :

(13) [*Every one of the two hundred and twenty-five*] *participants received the proceedings of the conference.*

le déterminant composé *every one of the two hundred and twenty-five* contient le numéral composé souligné qui peut être vu comme unité remplaçable par tout autre numéral cardinal (sauf *one* et *two*). Ainsi, nous obtenons l'expression productive *every one of the <Dnum>*, où *Dnum* représente un déterminant numéral cardinal.

L'élargissement de la grammaire de déterminants numéraux exige entre autres l'analyse combinatoire complète de tous les déterminants simples (*all*, *no*, *such*, *the*, *these*, *what*, *whose*, *my*, etc.). Par exemple, le graphe *DetDetDnum* (Fig. 19) représente les numéraux cardinaux (Fig. 11) précédés d'un ou de deux déterminants simples (le graphe *Poss* contient les déterminants possessifs : *my*, *your*, *one's*, etc.). L'alignement de plus de trois déterminants simples avant un numéral, et de plus de deux après un numéral ne semble pas admissible. La description devient beaucoup plus compliquée pour les déterminants composés contenant des prépositions, comme *of* dans l'exemple (13), car les séquences à reconnaître peuvent être longues (e.g. *As many as twenty of those other fifty participants were absent.*).

Rappelons que l'étude complète de déterminants composés est très complexe et ses limites sont difficiles à tracer, car il est nécessaire de considérer :

- les fractions (*three and two-third*) ;
- certains adverbes (*surprisingly many*, *almost every*, *certainly not all*, etc.) ;
- des possibilités d'insertions de modifieurs à l'intérieur d'un déterminant composé (e.g. *the United State's first third-country detention center*) ;

- les unités de mesures (*25 percent*, *350 miligrams*, etc. ; voir aussi l'article de Matthieu Constant dans ce volume) ;
- d'autres déterminants dits nominaux, analysés en détails pour le français par BUVET 1994 (e.g. *un kilo de pommes*, *un verre de lait*, *une armée de touristes*).

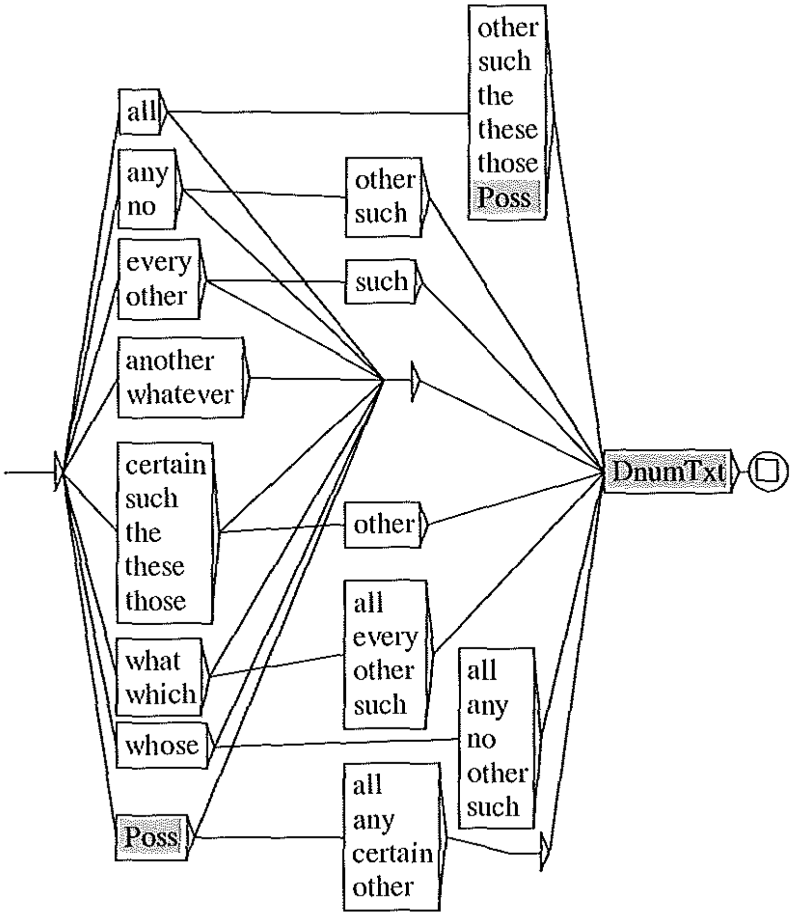

Figure 19. Graphe DetDetDnum *: déterminants numéraux précédés d'un ou de deux déterminants simples.*

7. Reconnaissance des numéraux par INTEX

Les automates et transducteurs finis présentés dans cette étude peuvent être employés dans le système INTEX en tant que dictionnaires *non ambigus* de mots composés[1]. (voir le manuel INTEX : SILBERZTEIN 1999–2000). Cela veut dire que chaque séquence reconnue sera considérée comme étant effectivement une occurrence d'un numéral dans le texte. De plus, seulement les séquences les plus longues seront prises en compte, par exemple dans la phrase (13) seul le numéral maximal *two hundred and twenty-five* sera étiqueté, et non pas ses sous-séquences : *two hundred, two hundred and twenty,* et *twenty-five.*

Les séquences reconnues par les graphes apparaissent alors dans le dictionnaire des mots composés du texte[2]. En annexe, nous présentons un extrait des cardinaux composés reconnus dans un grand corpus journalistique.

Dans la version actuelle d'INTEX (4.21) la définition du mot composé exige que son premier caractère soit obligatoirement une lettre de l'alphabet. Cette mesure, introduite pour des raisons opérationnelles, ne permet pas la reconnaissance, en tant que mots composés, de certains numéraux décrits plus haut. Par exemple, les graphes *DnumDig* (Fig. 1) et *DnumOrdDig* (Fig. 17) ne peuvent pas être employés sous INTEX en tant que dictionnaires de mots composés. Aussi, les branches supérieures des graphes comme ceux des figures 7, 9 et 10 ne sont jamais activées lors de l'analyse lexicale, donc les séquences comme *7.6 thousand* ne peuvent pas être reconnues.

8. Conclusion

Les outils à états finis sont très bien adaptés à la description des expressions numériques : 24 automates et 43 transducteurs, construits à partir de près de 70 mots simples, décrivent 10^{18} numéraux cardinaux et autant de numéraux ordinaux.

La taille des graphes obtenus est d'environ :

– 130 états et 1 500 transitions pour les automates

[1] Menu *Text→ Apply Lexical Resources.*

[2] Certaines de ces séquences sont pourtant des mots simples, car il nous a semblé plus cohérent de décrire tous les numéraux par une seule bibliothèque de graphes. Si nous voulions séparer les numéraux simples (comme *ninety, hundredth*) des numéraux composés (comme *ninety-one, two hundredth*), les graphes deviendraient beaucoup plus complexes et moins intuitifs.

– 2 700 états et 11 000 transitions pour les transducteurs.

Le temps de leur application à un corpus de plusieurs mégaoctets, comme un mois du journal *Herald Tribune*, est de l'ordre d'une quinzaine de minutes.

Bibliographie

BUVET (Pierre-André) : 1994, « Détermination : les noms », *Lingvisticae Investigationes*, XVIII :1 (Amsterdam : John Benjamins).

CHROBOT (Agata) : 1999, « Flexion automatique des mots composés », dans LAMIROY (Béatrice), KLEIN (Jean), PIERRET (Jean-Marie), eds., *Cahiers de l'Institut de Linguistique de Louvain. Actes du XVI^e Colloque Européen sur les lexiques et la grammaire comparés des langues romanes*, Louvain-la-Neuve, septembre 1997 (Louvain-la-Neuve : Peeters).

GROSS (Maurice) : 1997, « The Construction of Local Grammars », dans ROCHE (Emmanuel), SCHABES (Yves), eds., *Finite-State Language Processing* (Cambridge, Massachusetts : MIT Press).

MAUREL (Denis) : 1989, *Reconnaissance de séquences de mots par automates. Adverbes de dates du français*. Thèse de doctorat (Université Paris 7).

OALDCE : 1989, *Oxford Advanced Learner's Dictionary of Current English*, A.S. Hornby (Oxford University Press).

SILBERZTEIN (Max) : 1990, « Le dictionnaire électronique des mots composés », *Langue Française*, 87 (Paris : Larousse).

SILBERZTEIN (Max) : 1993, *Dictionnaires électroniques et analyse automatique de textes. Le système INTEX* (Paris : Masson).

SILBERZTEIN (Max) : 1997, *Intex 3.4. Reference Manual* (Paris : LADL).

SILBERZTEIN (Max) : 1999–2000, *Intex* (Paris : ASSTRIL).

Annexe

Numéraux cardinaux composés anglais reconnus par le graphe *Dnum* (Fig. 16) dans 8 Mégaoctets de texte de *Herald Tribune* (septembre 1994) :

eight million,8000000.DET+Num :p
Fifty-eight,58.DET+Num :p
Fifty-seven,57.DET+Num :p
Fifty-two,52.DET+Num :p
Five hundred,500.DET+Num :p
five billion,5000000000.DET+Num+US :p
five billion,5000000000000.DET+Num+GB :p
five hundred,500.DET+Num :p
Forty-one,41.DET+Num :p
forty-five,45.DET+Num :p
four million,4000000.DET+Num :p
nine million,9000000.DET+Num :p
Ninety-five,95.DET+Num :p
Ninety-two,92.DET+Num :p
ninety-four,94.DET+Num :p
One hundred,100.DET+Num :p
One thousand,1000.DET+Num :p
one billion,1000000000.DET+Num+US :p
one billion,1000000000000.DET+Num+GB :p
one million,1000000.DET+Num :p
seven million,7000000.DET+Num :p
Seventy-seven,77.DET+Num :p
Seventy-six,76.DET+Num :p
Seventy-three,73.DET+Num :p
six million,6000000.DET+Num :p
Sixty-eight,68.DET+Num :p
Sixty-five,65.DET+Num :p
Sixty-four,64.DET+Num :p
Ten million,10000000.DET+Num :p
Thirty- eight,38.DET+Num :p
Thirty-eight,38.DET+Num :p
Thirty- three,33.DET+Num :p
Thirty-Two,32.DET+Num :p
Three million,3000000.DET+Num :p
three billion,3000000000.DET+Num+US :p
three billion,3000000000000.DET+Num+GB :p
three hundred,300.DET+Num :p
three million,3000000.DET+Num :p
three trillion,3000000000000.DET+Num+US :p [1]
Twelve hundred,1200.DET+Num :p
Twenty-five,25.DET+Num :p
Twenty million,20000000.DET+Num :p
Twenty-nine,29.DET+Num :p
Twenty- One,21.DET+Num :p
Twenty-One,21.DET+Num :p
Twenty-one,21.DET+Num :p
Twenty-seven,27.DET+Num :p
Twenty-six,26.DET+Num :p
Twenty-two,22.DET+Num :p
twenty-four,24.DET+Num :p
twenty-one,21.DET+Num :p
Two billion,2000000000.DET+Num+US :p
Two billion,2000000000000.DET+Num+GB :p
Two million,2000000.DET+Num :p
Two thousand,2000.DET+Num :p
two hundred,200.DET+Num :p
two million,2000000.DET+Num :p

[1] La variante britannique du même numéral n'a pas été reconnue, car notre description ne dépasse pas 10^{18}, soit *one trillion* pour les nombres britanniques.