Une description morphologique structurée en arbre du verbe akkadien qui utilise des structures de traits et des transducteurs multirubans

François Barthélemy

Conservatoire National des Arts-et-Métiers (CNAM-CEDRIC)
292, rue Saint-Martin, 75003 Paris

francois.barthelemy@cnam.fr

Institut National de Recherche en Informatique (INRIA, projet Alpage)
domaine de Voluceau, 78153 Le Chesnay cedex

RÉSUMÉ. *Cet article est consacré à une grammaire du verbe akkadien utilisant des techniques de machines finies à états. Elle repose sur des techniques innovantes permettant de relier différentes représentations d'une forme (quatre dans cette grammaire) au moyen d'une structure arborescente et de compiler statiquement des structures de traits dans des transducteurs finis.*

ABSTRACT. *This article is devoted to a grammar of the Akkadian verb using finite state technology. It is based on new techniques for which relationships between several representations of a form (four in the Akkadian grammar) are expressed using a tree structure. Feature structures compiled statically in finite transducers are also involved.*

MOTS-CLÉS : *akkadien, morphologie, machines finies à états, structures de traits.*

KEYWORDS: *Akkadian, Morphology, Finite-State Machines, Feature Structures.*

1. Introduction

L'akkadien est la langue des anciens Babyloniens et Assyriens. Elle a été écrite en écriture cunéiforme, un système inventé à l'origine par les Sumériens qui avaient une autre langue. L'akkadien a été écrit pendant à peu près vingt-trois siècles en Mésopotamie et dans tout le Proche-Orient. La majeure partie des documents est constituée de tablettes d'argile.

L'akkadien est une langue sémitique. Sa morphologie comporte beaucoup de traits communs avec les autres langues sémitiques. Elle a aussi quelques originalités au niveau de la vocalisation et de la structure des schèmes. Sa flexion verbale est très riche.

Dans cet article, nous présentons une grammaire de la morphologie verbale de l'akkadien utilisant une approche à états finis. Cela s'inscrit dans une tradition bien ancrée qui remonte à la fin des années 1980 avec les travaux précurseurs de (Kay, 1987), suivis de beaucoup d'autres. Tout en bénéficiant des apports de ses devanciers, ce travail est original parce qu'il fonde l'analyse sur une description des formes fléchies structurée en arbre au lieu d'une structure linéaire.

Nous utilisons une classe de relations rationnelles qui explicite une structure au moyen d'opérateurs de produit cartésien typés. Ces relations peuvent être n-aires, c'est-à-dire consister en ensembles réguliers de n-uplets et non nécessairement de paires. Une relation n-aire peut se compiler automatiquement en un transducteur fini à n rubans.

La grammaire du verbe akkadien est une relation quaternaire décrite en deux parties distinctes : une première partie structurelle décrit la structure d'une forme au moyen d'un ensemble de contraintes décrites séparément et appliquées simultanément. Une seconde partie décrit les transformations de surface au moyen d'un ensemble de règles de réécriture appliquées en cascade. Cette partie décrit notamment la forme des verbes faibles. Ces verbes ont certains éléments de leur racine qui n'apparaissent pas dans les formes fléchies.

L'intérêt de notre grammaire n'est pas dans la réalisation d'une chaîne de traitement automatique du babylonien, mais dans la formulation de questions de recherche et la validation de certaines techniques originales mises en œuvre. Quant à un emploi en pratique, on pourrait envisager un usage pédagogique, en complément des manuels ainsi qu'une aide à l'identification des racines pour les débutants. Cette identification est notamment un préalable pour la consultation d'un dictionnaire. Dans certains cas (verbes I-faibles), il faut rechercher une entrée à une lettre qui ne figure pas dans la forme fléchie rencontrée. Dans d'autre cas, la lettre apparaît dans la forme, mais pas en position initiale. La tâche de reconnaissance de la racine est donc essentielle et n'a rien d'évident.

Dans la prochaine section nous présentons brièvement les relations multigrains utilisées par la grammaire, ainsi que le système `Karamel` qui met en œuvre ce modèle. Ce système comprend notamment un langage de définition de relations rationnelles

n-aires. La section 3 présente rapidement l'akkadien et sa morphologie verbale. Vient ensuite la description de la grammaire. Les deux dernières sections traitent des apports de ce travail et le comparent à d'autres grammaires de langues sémitiques en morphologie à états finis.

2. Les relations multigrains et le système Karamel

2.1. *Morphologie à états finis et relations multigrains*

Les *machines finies à états* – automates et transducteurs finis – sont des modèles opérationnels séduisants parce qu'elles sont efficaces et leur sémantique est claire. Grâce à leur contre-partie déclarative, les expressions régulières, elles forment un outil très utilisé en dépit de leur faible pouvoir expressif. Des algorithmes permettent de combiner ces machines (union, concaténation, intersection, composition) et de les optimiser (déterminisation, minimisation).

La morphologie à états finis est l'approche qui consiste à utiliser des machines finies pour réaliser la description morphologique des langues. Une telle description est exécutable et peut réaliser aussi bien l'analyse morphologique d'une forme fléchie que la génération d'une forme à partir de différents facteurs tels que le lemme et des traits morphologiques. Le modèle permet de représenter des analyses ambiguës de façon compacte grâce à un partage de structure.

La morphologie à états finis s'est développée au cours des années 80 et a connu de nombreux succès pour décrire la morphologie de langues appartenant à différentes familles, avec différents mécanismes de dérivation et de flexion : des langues utilisant des préfixes et suffixes en petit nombre (pour une forme donnée), telles que les langues indo-européennes, des langues agglutinantes telles le turc et des langues à morphologie non concaténative telles l'arabe.

La morphologie d'une langue est décrite généralement au moyen d'un formalisme à base de *règles contextuelles* et compilée en un transducteur fini qui met en relation formes de surface et représentations plus ou moins abstraites.

La morphologie à états finis connaît deux courants majeurs et des variantes. Le premier courant consiste à décrire par les règles des contraintes qui s'appliquent simultanément. C'est la *morphologie à deux niveaux* (Koskenniemi, 1983) qui a eu un grand succès mais semble en perte de vitesse depuis une décennie. Le second courant décrit des contraintes successives, s'appliquant dans un ordre déterminé. Ce sont les *règles de réécriture* (Kaplan et Kay, 1994). Ce modèle semble avoir pris le dessus, notamment grâce à une bonne implémentation, les *Xerox Finite-State Tools* (XFST en abrégé) (Beesley et Karttunen, 2003). En termes de calcul, le respect de contraintes simultanées peut s'exprimer par une intersection de transducteurs représentant chacun une contrainte et les contraintes successives comme une composition de transducteurs.

La morphologie multigrain a été proposée par l'auteur de cet article dans la lignée du modèle à contraintes simultanées et plus particulièrement dans celle de la

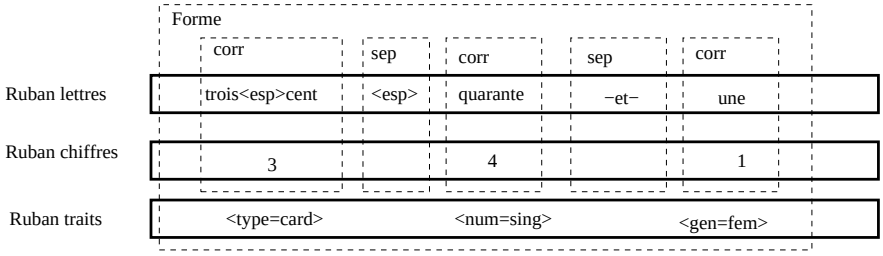

Figure 1. *Un exemple d'analyse multigrain*

morphologie à partition (Grimley-Evans *et al.*, 1996 ; Kiraz, 2001). Elle repose sur les *relations rationnelles multigrains* (Barthélemy, 2007b) qui sont des relations à *n* composantes, *n* pouvant être supérieur à deux, compilées en des transducteurs multi-rubans.

Comme des relations rationnelles n-aires, les relations multigrains définissent des n-uplets dont chaque composant est une représentation d'une forme donnée – représentation de surface, abstraite ou intermédiaire. Chaque représentation est une chaîne de caractères. Par exemple, il est possible de mettre en relation trois représentations des nombres : avec des chiffres, en toutes lettres, avec des traits morphologiques qui déterminent le caractère ordinal ou cardinal du nombre, son nombre grammatical (singulier ou pluriel), son genre (féminin, masculin). Un triplet d'une telle relation ternaire est :

(341, trois cent quarante et une, [type=card,num=sing,gen=fem])

Les relations multigrains permettent de mettre en relation terme à terme des sous-chaînes de ces différentes représentations. On peut exprimer le fait que le chiffre 3 *correspond*, dans un certain sens, au terme `trois cent`. Cela se fait au moyen d'unités d'analyse verticales appelées des *grains*. On a ainsi deux axes d'analyse : horizontal, les composantes de la relation correspondant à des rubans d'un transducteur, et vertical, les grains, des sous-unités concernant un sous-ensemble non vide des rubans de la relation. Les grains sont un moyen de synchroniser partiellement les différents rubans.

La figure 1 donne un exemple d'utilisation de grains pour le triplet donné ci-dessus. Elle illustre le fait que les grains sont typés : il y a un type `corr` pour les unités de correspondance lettres-chiffres, un type `sep` pour des éléments de jonction entre composants sur le ruban lettres et le type `forme` pour désigner la forme entière. Notons que d'autres découpages en grains auraient été aussi légitimes sur cet exemple : par exemple, on aurait pu ne pas distinguer d'éléments de jonctions. On aurait également pu mettre en relation les traits avec le seul élément final puisque la cardinalité, le nombre et le genre sont notés en fin de forme seulement.

Dans la version la plus simple des relations multigrains, celle que nous utilisons dans cet article, les différentes unités verticales doivent être imbriquées et la concaténation n'est possible qu'entre deux unités comportant les mêmes rubans. Ces relations

sont closes sous opérations rationnelles et aussi sous intersection et différence ensembliste.

2.2. *Le système Karamel*

Le système Karamel a été développé par l'auteur de cet article pour mettre en œuvre la morphologie multigrain. Il propose un langage pour définir les relations, ainsi qu'un environnement de développement qui offre une interface graphique pour écrire, compiler et tester des relations. Nous allons présenter ici les caractéristiques générales du langage. Des exemples concrets et des détails de syntaxe sont donnés dans la section décrivant la grammaire de l'akkadien.

Une grammaire en Karamel comporte une section de déclarations suivie d'une séquence de définitions de relations multigrains. La section de déclaration comporte différentes clauses qui définissent l'univers considéré, à savoir les symboles que l'on va manipuler, les composantes des relations, les grains et leur composition.

Les symboles sont structurés en *classes* qui sont des ensembles finis définis par extension et qui ont un nom. Un symbole est décrit au moyen d'un ou plusieurs caractères unicodes. Lorsqu'il y a un doute sur les limites d'un symbole ou sur le fait qu'un caractère est un symbole, la séquence de caractères peut être encadrée des symboles < et >. Par exemple, une voyelle longue est écrite par redoublement du caractère notant cette voyelle : `aa` pour un `a` long. Mais la séquence `aa` dans une expression régulière dénote la succession de deux `a` courts. Dans ce contexte-là, le caractère devra s'écrire `<aa>`. La notation avec chevrons s'impose pour tous les symboles qui ne sont pas composés exclusivement de lettres et de chiffres. Par exemple, un espace est noté `< >`, une virgule `<,>` et la chaîne vide `<>`.

Un symbole peut appartenir à plusieurs classes. Dans la définition du contenu d'une classe, le nom d'une autre classe peut apparaître comme abréviation pour désigner tous ses éléments. Un nom de classe peut de même être utilisé dans une expression régulière pour dénoter la disjonction de ses éléments.

Les relations sont des n-uplets sans distinction *a priori* entre entrée, sortie et niveau intermédiaire. La seule notion est celle de composante. Si l'on considère une relation comme un ensemble de n-uplets, une composante correspond à une position de ces n-uplets. Mais en fait, Karamel utilise des relations à composantes nommées, comme les bases de données relationnelles. Si l'on considère une vision tabulaire des relations, une composante est une colonne de la table, ce que l'on nomme un *attribut* dans la terminologie des bases de données relationnelles. Une composante est identifiée par son nom. Les composantes doivent être déclarées au moyen d'une clause `tape` avec leur nom et l'alphabet des symboles susceptibles d'apparaître dans les chaînes de la composante.

Les différents types de grains doivent être déclarés avec une clause `grain`. Ces grains sont des opérateurs de produit cartésien permettant d'agréger des sous-relations

indépendantes dans une nouvelle relation. Ils sont caractérisés par un nom, une arité et pour chaque membre, un nom, un type et une valeur par défaut. Pour bien distinguer deux notions, nous allons introduire une terminologie explicite : nous appellerons *ruban* une composante d'une relation, ce qui correspond à un niveau ou un étage dans les autres modèles de morphologie à états finis et à la notion de ruban (ou *bande*) d'un transducteur. Nous appellerons *champ* un paramètre d'un produit cartésien n-aire.

Comme les rubans des relations, les champs des grains sont nommés et ils ont chacun une valeur par défaut. Chaque type de grain a un nom. Supposons qu'un type de grain est défini avec comme nom `tg1` et comme champs `c1`, `c2`. Dans une expression régulière, on notera un grain de la façon suivante : `{tg1: c1 = w1, c2 = w2}`. Une notation positionnelle est possible, dans laquelle les valeurs des champs sont données dans l'ordre de la déclaration du type : `{tg1: w1, w2}`. Lorsque l'on utilise la notation avec noms, on n'est pas obligé de respecter cet ordre. Si certains champs sont omis dans un grain, implicitement ces champs contiendront leur valeur par défaut. Une notation spéciale désigne un grain ne comportant que les valeurs par défaut : `{tg1}`.

Karamel implémente des structures de traits non récursives. Les structures de traits sont typées. Les types doivent être déclarés dans la section déclaration d'une grammaire. Une structure de traits peut apparaître n'importe où dans une expression régulière, mais généralement les structures de traits apparaissent sur des rubans dédiés. Elles sont compilées statiquement. Il faut les utiliser avec précaution parce qu'elles permettent de décrire des dépendances à longue distance qui sont coûteuses et peuvent provoquer une explosion combinatoire. Les techniques de compilation utilisées sont décrites dans (Barthélemy, 2007a).

Un type de structure de traits est défini avec un nom et une liste de traits, avec pour chacun l'ensemble fini de symboles qu'il peut prendre comme valeur. Les valeurs sont des symboles ordinaires, qui doivent être déclarés. La notation des traits est la notation habituelle, sauf que le type de la structure doit être donné au début : `[Name:gen=masc,num=2]`. Comme c'est l'usage, il est possible de ne spécifier qu'une partie des traits et leur ordre n'est pas significatif. Le nom du type dénote une classe de symboles qui regroupe toutes les valeurs possibles pour les traits et les symboles auxiliaires utilisés dans la compilation des structures.

La langage Karamel fournit des macros appelées *abréviations*. Une abréviation est une notation pour un type de grain déjà déclaré où une partie des valeurs des champs est définie à la déclaration de la macro et une autre partie est définie à l'appel, sous forme de paramètres. La notation d'une abréviation est identique à celle d'un grain.

La langage Karamel offre trois façons de définir une relation multigrain : avec une expression régulière, avec un calcul ou avec une règle contextuelle. Nous allons voir ces trois types de définitions successivement.

Une expression régulière utilise les symboles, classes de symboles et grains définis dans les déclarations. Elles peuvent comporter les opérateurs rationnels et les extensions habituelles, comme par exemple l'optionnalité notée avec un point d'interrogation. De plus, l'intersection et la différence sont disponibles. Les opérateurs binaires

ne peuvent être utilisés que sur des expressions portant sur le même sous-ensemble des rubans de la relation. Il est par exemple possible de concaténer deux petits grains ou deux gros grains, mais pas un petit et un gros.

Le deuxième moyen de définir une relation est d'appliquer un opérateur à une ou plusieurs relations déjà définies. Tous les opérateurs utilisés dans les expressions régulières sont disponibles, mais il y en a trois autres qui sont spécifiques à ce deuxième type de définition. Il y a la *projection* qui supprime un ou plusieurs rubans de son opérande. Le deuxième opérateur est le *produit externe* qui combine une relation multigrain et un langage sur un ruban donné. Il est utilisé pour *appliquer* un transducteur sur une entrée qui n'est pas encore divisée en grains. Tous les partitionnements en grains de cette entrée sont d'abord calculés, puis il y a calcul de l'intersection de ces partitionnements avec un ruban de la relation. L'opération duale est la *projection externe* qui extrait un langage d'un ruban d'une relation. Une projection standard est d'abord effectuée, puis les informations concernant les limites de grains sont enlevées.

Les règles contextuelles sont un troisième moyen de définir une relation. Ces règles sont des règles de *restriction généralisée* proposée par (Yli-Jyrä et Koskenniemi, 2004). Ce sont une généralisation des règles à deux niveaux (règles de restriction de contexte ou de coercition de surface). Une règle consiste en trois expressions régulières : un univers, un motif gauche et un motif droit. Il s'agit d'une règle de type *si…alors…* : tous les n-uplets de l'univers qui concordent avec le motif gauche doivent aussi concorder avec le motif droit. Un symbole spécial noté # peut être utilisé dans les motifs pour identifier des positions ou des occurrences de symboles spécifiques qui doivent être communs aux deux motifs. Cela permet de définir l'équivalent de ce que l'on appelle le *centre* dans les autres sortes de règles contextuelles.

Les expressions régulières et les règles contextuelles peuvent comprendre des variables qui prennent leurs valeurs dans des ensembles finis de symboles. Une expression avec une telle variable est équivalente à la disjonction des expressions obtenues en substituant une valeur à la variable. Les variables permettent d'écrire l'unification de traits.

Une relation régulière peut décrire la *réécriture* d'un ruban. Les autres rubans peuvent éventuellement conditionner cette réécriture. Par exemple, une distinction par cas est susceptible de s'opérer en fonction de la valeur d'un trait. La réécriture s'exprime au moyen de couples (sous-chaîne avant, sous-chaîne après) au niveau le plus profond de l'arborescence. Une telle relation de réécriture peut être exprimée au moyen d'une expression régulière, d'un calcul ou d'une règle contextuelle. L'opération de réécriture consiste à appliquer cette relation de réécriture sur une relation ordinaire et se traduit par la modification du contenu du ruban concerné.

3. Présentation de l'akkadien et de sa morphologie verbale

3.1. *La langue et son écriture*

L'akkadien est une des langues de la Mésopotamie ancienne (Heise, 1995 ; Sanchez, 2005). Elle a été écrite dans une période courant approximativement de – 2300 à l'an 0, d'abord par des populations akkadophones, puis en tant que langue d'échange et de culture par des populations ayant une autre langue. Elle a notamment servi de langue diplomatique pour tout le Proche-Orient, y compris l'Égypte et l'Anatolie. Elle a été la langue administrative des empires babyloniens et assyriens.

L'akkadien est une langue morte depuis plus de 2000 ans. Elle est connue à travers les textes écrits qui ont été retrouvées en grand nombre dans des fouilles archéologiques, surtout sous forme de tablettes d'argile. Il existe des centaines de milliers de documents dans les musées et de nouvelles découvertes viennent continuellement enrichir ce stock. La langue est décryptée depuis la fin du XIX^{e} siècle.

L'akkadien était écrit au moyen de l'écriture cunéiforme empruntée au sumérien, une autre langue de la Mésopotamie. C'est une écriture hétérogène, avec un sous-système phonétique syllabique qui a été adapté à la phonologie de l'akkadien, et deux sous-systèmes sémantiques (logogrammes et déterminatifs) empruntés sans grands changements.

Dans le sous-système syllabique, chaque signe note phonétiquement une syllabe qui comprend nécessairement une voyelle, éventuellement précédée ou suivie d'une consonne. Les motifs représentés sont donc : V, CV, VC et CVC. Il est impossible d'écrire une consonne sans une voyelle immédiatement adjacente, comme par exemple une séquence de trois consonnes successives en milieu de forme ou une séquence de deux consonnes successives en position initiale ou finale. On peut supposer que certaines des voyelles présentes dans l'écriture n'étaient pas prononcées mais servaient seulement de support pour écrire une consonne. Notons au passage que contrairement à la majorité des langues sémitiques, toutes les voyelles sont écrites. La longueur des voyelles, bien que significative, n'est pas notée dans l'écriture.

3.2. *Racines et schèmes*

L'akkadien est rattaché à la famille des langues sémitiques. Il forme le principal et presque unique représentant de la branche orientale. La morphologie en général et celle du verbe en particulier sont typiques des langues sémitiques.

Une des spécificités de la morphologie des langues sémitiques est l'existence d'unités morphologiques discontiguës, c'est-à-dire d'éléments séparés par différents matériaux et qui constituent une unité d'analyse atomique. Au premier rang de ces unités est la *racine* d'une forme qui est constituée le plus souvent par trois consonnes que nous appellerons les *consonnes radicales*. Certaines racines moins nombreuses

ont deux ou quatre consonnes et certaines analyses peuvent admettre des racines comportant des voyelles.

Par exemple, la racine akkadienne prs est reliée au concept de découpage. Elle apparaît dans le verbe parāsu, qui signifie *couper* et *décider* (on pourrait dire aussi *trancher*), mais également dans le nom paras qui signifie *fraction* et dans le nom purussû, *décision*. Nous utiliserons cette racine pour la plupart des exemples que nous donnerons dans cet article. La notion de racine, identifiée depuis les premiers grammairiens de l'arabe classique, est toujours très utilisée et l'identification de la racine est une étape clé de l'analyse d'une forme.

Une forme fléchie comporte un élément central qui comporte lui-même notamment les trois consonnes radicales et que nous appellerons le *noyau verbal*. Autour de ce noyau, il y a des composants concaténatifs qui se traitent comme les affixes des langues indo-européennes. Ces composants sont des préfixes et suffixes notant des informations de genre, nombre, personne, cas (ex. : **ta**prus, pars**u**), des suffixes notant un mode grammatical, des clitiques tels que des pronoms suffixes (parras-**ki**) et des particules enclitiques diverses (**l**iprus).

Le système verbal comporte des formes conjuguées et des formes nominales. Les formes conjuguées opposent l'accompli à l'inaccompli. La distinction est plus de nature aspectuelle que temporelle, aussi emploierons-nous le terme *aspect* pour désigner cette information. Au-delà des deux aspects de base, il existe trois autres formes purement verbales : le parfait, utilisé pour exprimer un état intermédiaire (procès tout juste achevé ou postérieur à un accompli), l'impératif et le permansif qui exprime un état atemporel. Il y a trois formes nominales du verbe : l'infinitif, le participe actif et l'adjectif verbal. Ces formes sont soumises à déclinaison. Il y a deux genres, trois personnes, trois nombres – singulier, duel et pluriel – et trois cas – nominatif, accusatif et génitif. Nous appellerons ces formes également des aspects, bien que ce soit un usage abusif du terme.

L'akkadien comporte deux *modes grammaticaux* qui ne sont pas réellement des modes du verbe. Le *subjonctif* est utilisé dans les propositions subordonnées. Il est marqué par un suffixe *u* bref. Le *ventif* notait à l'origine une notion directionnelle qu'il a peu à peu perdue. Il est lui aussi noté par un suffixe (m, nim ou am).

3.3. *Morphologie du noyau verbal*

Différents éléments peuvent s'ajouter à la racine pour constituer le noyau du verbe akkadien. Parmi eux, il y a les voyelles (ex. : paris) en nombre variant de 0 à 3. Le noyau peut comporter des infixes qui suivent la première consonne du noyau, l'infixe t (pitrus) ou l'infixe tan (*pitanrus>pitarrus). Il peut y avoir un préfixe n ou š (ex. : šupris). Notons que ces préfixes sont considérés comme membres du noyau car leur présence est régie par les mêmes facteurs que les autres éléments du noyau et ils peuvent porter un infixe (ex. : šutapris). Le dernier type de composant du

	Voix I	Voix II	Voix III	Voix IV
Sous-voix 1	forme de base	factitif multiplicatif multiplicité d'objets	causatif factitif des verbes d'état	passif de la voix I incohatif
Sous-voix 2	réciproque réflexif	passif de voix II	passif de voix III	
Sous-voix 3	itératif habituel	itératif habituel	itératif habituel	itératif habituel

Tableau 1. *Sémantisme approximatif des schèmes*

noyau est constitué de la *gémination* ou redoublement d'une radicale, la deuxième ou la troisième (ex. : `purris`).

Les différentes transformations sont organisées en un système à deux dimensions quasiment orthogonales, chacune comportant des transformations mutuellement exclusives. Selon un premier axe, les formes sont réparties en quatre catégories. Suivant la terminologie de (Malbran-Labat, 2001), nous les appellerons *voix* et les numéroterons avec un nombre romain. La voix I ne comporte pas de transformation, la voix II une gémination, la voix III un préfixe `š` et la voix IV un préfixe `n`. L'autre dimension que nous appellerons *sous-voix* est numérotée en chiffre arabe. La sous-voix 1 ne comporte pas d'infixe, la sous-voix 2 un infixe `t` et la sous-voix 3 un infixe `tan`. N'importe quelle voix peut se croiser avec n'importe quelle sous-voix à l'exception de la voix IV qui ne permet pas de sous-voix 2. Il reste un total de 11 croisements entre voix et sous-voix qui forment ce que l'on nomme les *schèmes* ou les *thèmes* dans la grammaire des langues sémitiques.

Si le système de schèmes est facile à décrire en termes de structure morphologique, il est plus difficile d'en décrire le sémantisme. Une part non négligeable de la sémantique des voix est de nature lexicale. Le tableau 1 donne une idée approximative des aspects verbaux associés aux différentes voix.

3.4. *Vocalisation*

L'akkadien connaît quatre couleurs de voyelles – a, e, i, u – et deux longueurs – courte ou longue.

Dans d'autres langues sémitiques, il existe en plus de la racine un second élément discontigu dans la morphologie verbale, à savoir un schéma vocalique parfois composé de plusieurs voyelles intercalées dans la racine. Par exemple en arabe, les deux voyelles `a` sont nécessaires pour identifier le temps d'une forme comme `katabtu` (*j'ai écrit, accompli*). En akkadien, nous pensons qu'il n'existe pas de schémas de plus d'une voyelle et par conséquent, pas de discontiguïté.

La vocalisation des noyaux verbaux est un des points les plus complexes de la

Composant	Voix	Sous-voix	Aspect	Lexique
Gémination 1	X			
Gémination 2	X	X	X	
Infixe t 1		X		
Infixe t 2			X	
Infixe tn		X		
Préfixe mu	X	X	X	
Prefixes š et n	X			
Voyelle aspectuelle	X	X	X	
Voyelle lexicale	X	X	X	X

Tableau 2. *Association trait morphologique-composant morphologique*

morphologie akkadienne. Malbran-Labat distingue trois sortes de voyelles, selon les traits morphologiques qui les déterminent :

– certaines voyelles sont déterminées par la voix, la sous-voix et l'aspect. Nous les appellerons *voyelles aspectuelles* ;

– certaines voyelles sont déterminées par les trois mêmes traits plus une information lexicale : les racines verbales sont distribuées dans cinq classes différentes qui utilisent des voyelles différentes pour une même valeur des trois autres traits. Nous les appellerons *voyelles catégorielles*. Chaque classe est caractérisée par un couple de voyelles, la première utilisée notamment pour l'inaccompli, la seconde pour l'accompli. Les cinq classes sont `a/a`, `i/i`, `u/u`, `a/u` et `a/i` ;

– certaines voyelles ne sont pas déterminées par des traits morphologiques, si ce n'est indirectement. Leur couleur est déterminée par la consonne qui suit et leur présence a pour but d'éviter les séquences de trois consonnes successives (deux en position finale ou initiale). Nous appellerons ces voyelles des *voyelles d'appui*.

Une caractéristique de l'akkadien est qu'une seule des voyelles du noyau porte une information d'aspect et de schème. C'est la dernière voyelle du noyau sauf dans quelques cas au schème I.1 (participe et impératif) où c'est la première voyelle qui est significative.

La vocalisation du noyau comprend plusieurs sous-systèmes : les formes conjuguées opposent les voix I aux voix II et III, la voix IV étant partagée entre les deux groupes selon les aspects et sous-voix ; les formes nominales et pronominales pour leur part opposent le schème I.1 aux autres schèmes.

Le tableau 2 associe les différents composants pouvant apparaître dans le noyau à des traits morphologiques. Les deux infixes `t` peuvent se cumuler alors que les deux géminations sont exclusives l'une de l'autre.

Pour éviter les successions de trois consonnes, deux solutions sont utilisées : faire disparaître une des consonnes ou insérer une voyelle. La disparition de consonne se

produit avec une gémination marquant l'inaccompli (en aucun cas avec une gémination de voix II), le `n` de l'infixe `tan` ou une radicale faible (*cf.* sous-section 3.5). Dans les autres cas, une voyelle est insérée. L'insertion peut se faire après la première ou la deuxième consonne selon la nature morphologique des consonnes de la séquence, par exemple : `*iptras`>`iptaras` mais `*ušpras`>`ušapras`. La consonne choisie parmi les deux possibles peut se caractériser par un ordre de priorité : préfixe n < deuxième radicale < première radicale < préfixe š < infixe.

Les deux tableaux 3 donnent un aperçu incomplet de la richesse de la morphologie verbale de l'akkadien. Ils donnent pour la racine `prs` les formes conjuguées à la troisième personne singulier masculin (deuxième personne pour l'impératif) et les formes nominales au nominatif masculin singulier. Un tableau contenant toutes les formes ne tiendrait pas sur une page : il y en a plus de 900. Cette présentation a une limite notable : les formes conjuguées sont dépourvues de suffixes alors que certains phénomènes ne sont observables qu'en présence de suffixes.

3.5. *Verbes faibles*

Comme dans les autres langues sémitiques, il existe en akkadien des *verbes faibles*, c'est-à-dire des verbes qui ont une ou plusieurs consonnes radicales qui disparaissent dans certaines ou toutes les formes fléchies. Ces consonnes sont *aleph*, noté `'`, *jod* (j) et *wav* (w). La disparition d'une telle consonne laisse parfois une trace : changement de couleur (`*'apāšu` > `epēšu`) ou de longueur des voyelles (`*i'kul`> `īkul`), changement de place de la gémination (`*uba''alū` > `ubellū`). Il peut également y avoir une disparition sans aucune trace (`*ikla' > ikla`) ou un maintien de la consonne (`ukta''in`). Les verbes faibles sont relativement fréquents : approximativement une forme verbale sur deux est faible. Certaines racines peuvent comporter deux, voire trois, radicales faibles.

(Malbran-Labat, 2001) écrit : *les caractéristiques morphologiques sont dans une large mesure identiques pour les verbes forts et les verbes faibles [...] les particularités des verbes faibles sont peu nombreuses. C'est l'évolution phonétique, due à la nature de la faible, qui les différencie des verbes forts*.

La règle la plus productive est la suivante : en contact avec une voyelle, la faible disparaît en essayant d'allonger cette variable et éventuellement, de changer sa couleur. La tentative d'allongement échoue si cela viole certaines règles phonologiques générales de la langue : si la voyelle est dans une syllabe fermée ou si la syllabe suivante comporte déjà une longue.

4. Grammaire Karamel du verbe akkadien

Cette section est consacrée à une grammaire du verbe akkadien écrite en Karamel. Cette grammaire a différentes caractéristiques : elle utilise plus de deux rubans, elle est structurée au moyen d'une arborescence de profondeur deux, elle utilise des structures

Schème	Formes nominales nominatif masculin singulier			Forme pronominale masc. sing. 3
	Participe	Adjectif	Infinitif	Permansif
I.1	parisu	parsu	parasu	paris
I.2	muptarsu	pitrusu	pitrusu	pitrus
I.3	muptarrisu		pitarrusu	pitarrus
II.1	muparrisu	purrusu	purrusu	purrus
II.2	muptarrisu	putarrusu	putarrusu	putarrus
II.3	muptarrisu		putarrusu	putarrus
III.1	mušaprisu	šuprusu	šuprusu	šuprus
III.2	muštaprisu	šutaprusu	šutaprusu	šutaprus
III.3	muštaprisu		šutaprusu	šutaprus
IV.1	mupparsu	naprusu	naprusu	naprus
IV.3	muttaprisu		itaprusu	itaprus

Schème	Formes verbales			
	masculin singulier 3			masc. sing. 2
	Inaccompli	Parfait	Accompli	Impératif
I.1	ipar(r)as	iptar(r)as	iprus	purus
I.2	iptar(r)as	iptatras	iptarus	pitrus
I.3	iptanar(r)as	iptatarras	iptarrus	pitarrus
II.1	uparras	uptarris	uparris	purris
II.2	uptarras	uptatarris	uptarris	putarris
II.3	uptanarras	uptatarris	uptarris	putarris
III.1	ušapras	uštapris	ušapris	šupris
III.2	uštapras	uštatapris	uštapris	šutapris
III.3	uštanapras	uštatapris	uštapris	šutapris
IV.1	ipparas	ittapras	ipparis	napris
IV.3	ittanapras	ittatapras	ittapras	itapras

Tableau 3. *Résumé de la flexion de la racine* `prs`

de traits. La structure des formes est décrite au moyen de contraintes simultanées alors que certaines transformations de surface sont décrites au moyen de contraintes séquentielles.

Le choix entre contraintes simultanées et contraintes séquentielles s'est effectué sur la base suivante : les contraintes simultanées sont les seules permettant de décrire sans restriction les interactions entre différents rubans. La réécriture simultanée de plusieurs rubans n'est possible que sous des conditions qui ne sont pas simples à vérifier (Barthélemy, 2007c). Les règles de réécriture n'ont donc été utilisées que pour un seul ruban, pour exprimer des phénomènes complexes qu'il aurait été difficile d'exprimer en une seule fois. Les règles séquentielles cassent cette complexité en utilisant des formes intermédiaires, chaque étape de réécriture étant simple à décrire.

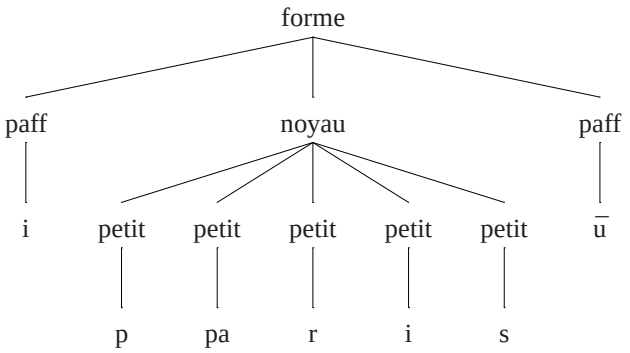

Figure 2. *Structure associée à la forme* ipparisū

4.1. *Structure des formes verbales*

La structure des formes verbales est décrite au moyen d'un arbre de profondeur deux. Le premier niveau de l'arbre est consacré à la partie concaténative de la grammaire : il décrit la façon dont les différents affixes et le noyau se combinent de façon cohérente. Cette combinaison distingue trois couches successives autour du noyau : les affixes dits *personnel* qui dépendent du genre, du nombre et de la personne ; les suffixes de modes ; les clitiques. Le second niveau de l'arborescence détaille la formation du noyau avec la racine et les différents éléments intercalés dedans.

La structure des formes verbales est exprimée en Karamel au moyen de cinq types de grains différents. Chacun de ces grains associe une représentation de surface à une structure de traits qui est la forme abstraite correspondante. Les grains du premier niveau de l'arbre, que nous appellerons désormais *les gros grains*, sont décrits au moyen de quatre types de grains différents : un pour le noyau, un pour les affixes personnels (type `paff`), un pour les suffixes de mode et un pour les clitiques. Le deuxième niveau de l'arborescence utilise un seul type de grain, le même pour toutes les composantes du noyau nommé `petit` (pour *petit grain*).

Chaque type de grain utilise un type de structure de traits différent qui contient les traits morphologiques concernés par la portion de forme décrite. Par exemple, la structure de traits décrivant les affixes personnels contient les quatre traits personne, genre, nombre et aspect, alors que la structure de traits associée à un grain de type `noyau` comporte les traits voix, sous-voix, aspect et classe lexicale. En ce qui concerne les petits grains, la structure de traits note la nature du composant (par exemple radicale, infixe t, etc). La figure 2 donne un exemple de structure associée à une forme.

On voit sur ce schéma que les types de grains sont les étiquettes des nœuds internes de l'arbre. Ils jouent là le même rôle que les non-terminaux d'une grammaire non contextuelle.

Gros grain	paff	noyau					paff
ggfs	masc,pl,3	accompli,IV.1, classe a/u					masc,pl,3
pgfs		pref. n	rad	rad.	voy. cat.	rad.	
lex	i	n	p	r	i	s	$\overline{u}$
surf	i	p	pa	r	i	s	$\overline{u}$

Tableau 4. *Exemple d'analyse d'une forme verbale*

La grammaire utilise quatre rubans distincts : un pour les structures de traits des gros grains (nom : `ggfs`), un pour les structures de traits des petits grains (nom : `pgfs`), un pour la représentation de surface qui est la transcription de l'akkadien dans un alphabet latin étendu (nom : `surf`). En plus de ces trois rubans nécessaires, un autre ruban conserve une forme intermédiaire que nous appellerons *forme lexicale* (nom : `lex`), qui est l'équivalent de la représentation lexicale en morphologie à deux niveaux : une représentation canonique des composantes indépendante du contexte. Cette représentation intermédiaire n'est pas absolument nécessaire, mais elle est informative pour éclairer une analyse et de plus elle est fort utile au cours du développement de la grammaire, lors de sa mise au point.

Le tableau 4 offre un exemple de représentation tabulaire de cette arborescence avec un aperçu des valeurs des traits. La distinction entre forme lexicale et forme de surface provient d'une assimilation du préfixe `n` à la radicale `p` et de l'ajout d'une voyelle d'appui après la première radicale.

Avec la syntaxe Karamel, la structure du tableau 4 s'écrit :

```
{forme:
  {paff: [pfs:gen=masc,pers=3,num=pl], i},
  {noyau: [nfs:voix=IV,sous=1,asp=accompli,lcat=a_u],
    {petit: [sfs:typ=rad], n, p}
    {petit: [sfs:typ=ifx_t], p, pa}
    {petit: [sfs:typ=rad], r}
    {petit: [sfs:typ=vcat], i}
    {petit: [sfs:typ=rad], s}}
  {paff: [pfs:gen=masc,pers=3,num=pl], <uu>, <uu>}};
```

Cette écriture est une expression régulière Karamel qui suppose la déclaration préalable des symboles, types de structures de traits, rubans et types de grains utilisés. Pour chacune de ces construction, nous allons prendre l'exemple d'une déclaration.

```
class voy is a, e, i, u, <aa>, <ee>, <ii>, <uu>;
class let is <voy>, <cons>;
class voix is I, <II>, <III>, <IV>;
fstruct nfs is [asp=<asp>,voix=<voix>,sous=<sous>,lcat=<lcat>]
tape surf: <let>;
```

```
grain noyau is {ggfs: ggfs = [nfs];
                sab: pgfs, lex, surf = {petit}*}
```

La déclaration des symboles et des valeurs de traits est identique : elle utilise la construction `class`. Un type de structure de traits a un nom (ici, `nfs`, les traits concernant le noyau verbal) et une série de noms de traits avec leur domaine de valeur qui est une classe de symboles préalablement définie. La déclaration du ruban donne son nom et son alphabet qui est également une classe de symboles. Le type de grain `noyau` a deux composantes nommées `ggfs` et `sab`. Pour chaque composante sont spécifiés outre le nom, les rubans qu'elle comporte et un domaine de valeurs décrit au moyen d'une expression régulière. L'expression régulière `[nfs]` dénote n'importe quelle structure de traits de type `nfs`, aucune valeur de traits n'étant spécifiée. L'expression régulière `{petit}*` dénote n'importe quelle séquence de grains du type `petit`, qui doit avoir été défini précédemment.

4.2. *Description du noyau verbal*

Le noyau verbal est construit autour d'une racine en utilisant des composants qui viennent d'un ensemble fini. Pour une forme donnée, chaque composant de l'ensemble peut apparaître ou non et il n'a au plus qu'une occurrence et une place possible. L'infixe `t` peut être doublé, mais on considère qu'il s'agit de deux composants distincts, qui ne sont pas déterminés par les mêmes traits (l'un est déterminé par l'aspect, l'autre par la sous-voix). Chaque composant est décrit dans la grammaire par une expression régulière qui décrit les cas où il est présent et ceux où il est absent. Comme la présence ou l'absence sont déterminées par les traits du gros grain `noyau`, l'expression régulière décrit le gros grain dans son ensemble. Voyons l'exemple de l'infixe `tan`.

```
regexp infixe_tan is
 {noyau: [nfs:sous=3],
   {petit}*{petit: [sfs:typ=<ifx_tan>], tan, tan}{petit}*};
 {noyau: [nfs:sous=1|2], {petit: [sfs:typ=<typ>-<ifx_tan>]}*};
end
```

Les deux premières lignes spécifient le cas où l'infixe est présent (la sous-voix est 3). La dernière ligne spécifie le cas où l'infixe est absent : il n'y a pas de petit grain de type `<ifx_tan>`. L'expression régulière nommée `infixe_tan` est la disjonction (union) de ces deux cas. Il n'y a ici que deux cas parce que la composante ne dépend que d'un trait. Pour des composantes déterminées par une combinaison de trois ou quatre traits, il y a jusqu'à une douzaine de cas différents.

Une autre expression régulière appelée `ordre_noyau` décrit l'ordre des composants. Une description de l'ensemble des noyaux correctement construits est obtenue par intersection de `ordre_noyau` avec les expressions régulières décrivant les composantes (une dizaine d'expressions).

```
noyau=intersect(ordre_noyau,infixe_tan,prefixe_voix,...);
```

4.3. *Concaténation des gros grains*

La section précédente explique comment l'on décrit l'ensemble des noyaux possibles au moyen d'une relation rationnelle. Nous allons aborder à présent la description des autres types de gros grains et la façon dont on les assemble, laquelle fait intervenir une dépendance à longue distance.

Les différents types de préfixes et de suffixes sont décrits chacun par une expression régulière qui est une énumération de toutes les valeurs possibles.

```
class fapref is accompli, inaccompli, parfait;
regexp prefixe_personnel is
  {paff: [pfs:pers=3,atyp=<pers>,asp=<fapref>],i,i};
  {paff: [pfs:pers=2,atyp=<pers>,asp=<fapref>],ta,ta};
  {paff: [pfs:pers=1,num=<sg>,atyp=<pers>,asp=<fapref>],a,a};
  {paff: [pfs:pers=1,num=<pl>,atyp=<pers>,asp=<fapref>],ni,ni};
end
```

Pour simplifier la formulation des règles de concaténation des gros grain, nous avons introduit des suffixes sans trace au niveau de surface, ce que l'on note parfois par l'affixe $\emptyset$. Cela évite de distinguer les cas avec suffixe et sans suffixe. Utiliser ou non ce type d'affixe relève d'un choix dont les critères sont de nature théorique et pratique. Dans notre grammaire, nous en avons utilisé pour les suffixes personnels et pas pour les petits grains. Au lieu de considérer qu'un type de petit grain peut être absent ou présent, on aurait pu supposer qu'il est toujours présent, mais parfois sans trace.

Les préfixes et suffixes personnels des formes conjuguées sont partiellement redondants parce qu'ils dépendent des mêmes traits : personne, genre et nombre. Ils sont parfois considérés comme un seul composant morphologique discontigu, composé avec le noyau par une opération de *circonfixation*. Dans la description Karamel, cette dépendance à longue distance entre préfixe et suffixe est traitée par l'intersection de deux descriptions : la première décrit la concaténation libre de préfixe, noyau et suffixe ; la seconde décrit l'égalité des traits communs aux préfixes et suffixes au moyen de variables.

Un autre type de dépendance entre gros grains concerne le type d'affixe personnel : il y a quatre types d'affixes différents déterminés par l'aspect de la forme. L'aspect est donc contenu dans les structures de traits des grains de type `paff` et `noyau`. Ici encore, l'égalité des différentes occurrences du trait est exprimée au moyen d'une variable. En Karamel, les variables sont notées par un nom débutant par la caractère $.

```
concatenation_grains=
   union(concat(noyau,suffixe_personnel),
         concat(prefixe_personnel,noyau,suffixe_personnel));
regexp dependances is
  {paff: [pfs:asp=$a,pers=$p,num=$n,gen=$g]}
    {noyau: [nfs:asp=$a]}
    {paff: [pfs:asp=$a,pers=$p,num=$n,gen=$g]};
  {noyau: [nfs:asp=$a]}{paff: [pfs:asp=$a]};
end
couche_1=intersect(concatenation_grains,dependances);
```

Les descriptions données ici distinguent deux cas : des formes avec préfixe personnel et des formes sans préfixe personnel. Le nom `couche_1` se réfère à la décomposition en trois couches : affixes personnels, suffixes de mode et clitiques. Les deux autres couches ne seront pas détaillées ici : leur description suit les mêmes principes que ceux exposés pour les affixes personnels.

4.4. *Contraintes séquentielles*

Nous avons vu jusqu'ici une description de la structure des formes verbales. Cette description construit une relation à quatre rubans dans laquelle les formes lexicales et de surface sont identiques. À partir de ce point de départ, la forme de surface est l'objet de réécritures successives au moyen de règles appliquées en séquence. Ces règles ne réécrivent que le contenu d'un des ruban et ne modifient pas la structure en grains. Les autres rubans néanmoins peuvent être utilisés dans les règles. Par exemple, la réécriture peut être déterminée par la valeur d'un trait.

La grammaire utilise la réécriture pour traiter trois phénomènes : l'insertion des voyelles d'appui, la disparition des radicales faibles et diverses transformations de surface telles que les assimilations.

La partie structurelle de la grammaire ne comporte aucune voyelle d'appui car ces voyelles ne sont pas déterminées par les traits morphologiques. Il n'y a donc pas lieu de les considérer comme un type de petits grains. Ces variables sont déterminées par des contraintes graphico-phonétiques (*cf.* section 3.4). Elles sont insérées au moyen de règles de réécriture dans le petit grain de la consonne qui précède. Les règles d'insertion sont complexes et ont pour but d'éviter les séquences de deux consonnes en initiale et finale et les séquences de trois consonnes en position médiane.

Une cascade de règles est utilisée pour définir l'ordre de priorité entre composants du noyau lorsqu'une voyelle d'appui est ajoutée (*cf.* sous-section 3.4). Une première règle ajoute une voyelle après un infixe s'il y a une séquence de trois consonnes comportant un infixe. Une deuxième règle fait la même chose pour le préfixe š et ainsi de suite pour chaque composant susceptible de porter une voyelle d'appui.

Les transformations diverses décrites dans la grammaire comprennent notamment la coloration en u de la première voyelle aux voix II et III, l'assimilation de la consonne n à d'autres consonnes, sauf pour des formes I-faibles à la voix IV et la disparition d'un n initial avant un infixe.

Les différents phénomènes décrits par des règles de réécriture peuvent avoir plusieurs occurrences dans une même forme. Dans ce cas, la description au moyen d'une règle contextuelle est plus pratique que l'utilisation d'une simple expression régulière. Les expressions régulières sont en revanche plus simples à écrire pour des phénomènes ayant au plus une occurrence dans une forme, comme c'est le cas pour les affixes et les composants du noyau.

4.5. *Prise en compte du lexique*

Nous n'avons pas abordé jusqu'ici la question du lexique. Nous utilisons dans la grammaire une liste de 799 racines akkadiennes provenant de (Breckwoldt *et al.*, 2000), qui ne précise ni la classe de vocalisation ni les voix attestées. Nous ne disposons pas d'une meilleure ressource. Compte tenu de l'imperfection de ce lexique, nous avons développé deux variantes de la grammaire en parallèle : l'une utilise la liste de racines et l'autre ne comporte aucun lexique. Les analyses sont alors des conjectures sur des racines possibles, à contrôler dans un dictionnaire.

La racine est un élément à prendre en compte dans la composition du noyau. Chaque racine est décrite comme un gros grain de type noyau où les composants autres que les radicales ne sont pas spécifiés. Pour faciliter l'écriture, une abréviation est définie.

```
abbrev racine is {r1 = <cons>; r2 = <cons>; r3 = <cons>}
   for {noyau: sab= {petit}* {petit: [sfs:typ=rad], @r1, @r1}
        {petit}* {petit: [sfs:typ=rad], @r2,@r2}{petit}*
        {petit: [sfs:typ=rad], @r3,@r3}}
regexp lexique is
  {racine: p, r, s};
  {racine: <aleph>, k, <sh>};
  ...
```

L'abréviation racine est un n-uplet à trois positions, une pour chaque radicale. Ces radicales sont substituées par macro-expansion aux symboles préfixés par le signe @ dans l'expression régulière suivant le mot-clé for. C'est ainsi que {racine: p, r, s}; dénote l'expression suivante :

```
{noyau: sab= {petit}* {petit: [sfs:typ=rad], p,p}
        {petit}* {petit: [sfs:typ=rad],r,r}{petit}*
        {petit: [sfs:typ=rad],s,s}}
```

5. Évaluation

5.1. *Couverture*

Au moment où ces lignes sont écrites, la grammaire couvre les verbes trilitères forts et les verbes avec une consonne faible. Les verbes à plusieurs radicales faibles ne font pas l'objet d'un traitement spécifique et nous ne savons pas dans quelle mesure ces verbes sont couverts par la grammaire. Le traitement des clitiques est embryonnaire.

La grammaire comporte 71 constructions `regexp`, 65 calculs et 6 règles contextuelles. Le transducteur représentant l'ensemble des formes verbales sans lexique a 496 429 états et 629 377 transitions. Avec le lexique, il y a environ 3 millions d'états et 3,5 millions de transitions. À titre de comparaison, une autre grammaire de l'akkadien (Kataja et Koskenniemi, 1988) comporte 123 règles à deux niveaux, une grammaire de l'arabe (Beesley, 1998a) comporte 66 règles de réécriture, une grammaire de l'hébreu (Wintner, 2008) se compile en un transducteur de 2 millions d'états et 2,2 millions de transitions.

La représentation la plus abstraite d'une forme est constituée par les structures de traits des gros grains qui contiennent des traits de nature morphologique. Cette représentation est néanmoins relativement superficielle : elle contient l'information à propos du schème sans en donner une interprétation sémantique. Ainsi, il y a des traits notant la voix et la sous-voix, mais il n'y a pas de trait notant le fait qu'une forme soit passive ou factitive. La raison de ce fait est que l'interprétation sémantique des schèmes échappe en grande partie à l'analyse morphologique et relève des niveaux syntaxique, pragmatique et sémantique.

Le développement et la mise au point de la grammaire ont été conduits en utilisant un jeu de tests artificiel, à savoir le contenu de certains tableaux de conjugaison donnés dans différents ouvrages (Malbran-Labat, 2001 ; Huehnergard, 2005 ; Buccellati, 1996). Un tel tableau comporte de nombreuses formes d'un petit nombre de racines (entre une et quatre). La description ayant été d'une certaine façon spécialisée pour ce jeu de tests, il ne permet pas de mesurer la couverture de la grammaire.

Le système Karamel permet d'automatiser la réalisation des tests, ce qui est fort utile dans la phase de mise au point : chaque modification de la grammaire est susceptible d'introduire des perturbations. Des tests peuvent être associés aux différentes relations décrites et la réalisation des tests de non-régression se fait littéralement en activant un bouton.

Par ailleurs, nous avons constitué manuellement un petit corpus de formes verbales recueillies dans le code de Hammurabi qui comporte quelques dizaines de formes verbales. Ce n'est évidemment pas un corpus très représentatif et les résultats obtenus ne sont donc pas très significatifs. Pour l'instant, nous n'avons pas d'indication de couverture véritablement crédible à faire valoir. La constitution d'un corpus et la

réalisation de tests ont un coût important alors que l'utilité pratique d'une analyse automatique n'a rien d'évidente.

La grammaire dans sa version sans lexique est à même d'émettre des hypothèses quant aux racines plausibles pour une forme donnée. Par ailleurs, pour un usage pédagogique, la grammaire peut facilement être enrichie de gloses explicatives pour expliciter et motiver ses analyses. Il suffit pour cela d'ajouter un ruban de plus dans la relation.

5.2. *Difficultés*

Il serait présomptueux de penser que notre travail apporte des connaissances nouvelles sur la langue akkadienne. Il vient confirmer certaines difficultés que les spécialistes connaissent bien et appeler des clarifications sur certains points.

Les difficultés concernent notamment la gémination d'inaccompli et l'infixe `tan`. Cette gémination est optionnelle pour les formes fortes et devient systématique dans certaines formes faibles. La géminée et le `n` de l'infixe ont un comportement très comparable. Le `n` en s'assimilant à la seconde radicale, ressemble à une gémination. Ces deux consonnes sont à moitié faibles : elles subsistent dans les séquences de deux consonnes mais tombent dans les séquences de trois consonnes, sauf dans le cas où il y a une radicale faible dans la séquence. Dans ce cas-là, c'est la radicale qui tombe.

La description du vocalisme de l'akkadien est un autre point délicat. Les ouvrages tendent à présenter la vocalisation de la racine au moyen de deux voyelles caractérisant l'aspect, la voix, la sous-voix et la racine. Ce système nous semble peu satisfaisant car il ignore certaines réalités. D'abord, le nombre de variables dans la racine n'est pas fixe. Il y en a entre une et trois. Le nombre n'est pas directement lié aux traits tels que l'aspect ou la voix, mais à des critères de prosodie et d'écriture. Ensuite, il n'y a qu'une voyelle qui dépende véritablement des quatre traits et c'est dans presque tous les cas la dernière. L'usage de présenter des schémas de deux voyelles est certainement emprunté à d'autres langues sémitiques ou au sémitique comparé.

Par ailleurs, la description des voyelles en trois catégories (aspectuelle, catégorielle, appui) a aussi des limites. Elle fonctionne assez mal pour l'impératif dont les voyelles sont catégorielles par leur couleur et voyelles d'appui par leur propension à disparaître.

En ce qui concerne l'emplacement des voyelles, nous avons utilisé un modèle additif où l'on part avec les seules voyelles significatives et où l'on insère les voyelles d'appui en utilisant un ordre de priorité. Une présentation inverse est souvent utilisée, où les formes comprennent initialement de nombreuses voyelles et une règle fait disparaître celles qui sont inutiles selon certains critères phonético-graphiques. Un essai d'implémentation de cette règle a donné des résultats catastrophiques, peut-être liés à une compréhension imparfaite du mécanisme décrit. Les grammaires abordent peu ce point qui n'a pas une grande importance pratique pour la lecture et l'analyse

des textes, puisque les voyelles d'appui ne portent pour ainsi dire pas d'information morphologique.

La grammaire de l'akkadien est actuellement la plus grosse développée avec des relations multigrains et en utilisant le système Karamel. Elle démontre la possibilité de traiter des problèmes de taille réelle avec ces technologies. Le lecteur pourrait douter de ce fait, en considérant qu'il s'agit seulement d'une description partielle d'une langue avec un lexique de petite taille. Le traitement est partiel, certes, mais il comporte la partie la plus difficile de la morphologie akkadienne, et celle-ci est très riche. Quant au lexique, Karttunnen a montré que contrairement à l'intuition, ce n'est pas un facteur de complexité, mais un moyen de limiter les risques d'explosion combinatoire (Karttunen, 1994).

Cette grammaire est également un exemple original d'approche hybride, qui utilise à la fois des contraintes simultanées et des contraintes successives. L'approche simultanée dans la lignée de la morphologie à deux niveaux et à partition, est utilisée pour décrire la morphotactique alors qu'une cascade de règles de réécriture est utilisée pour les alternances au sens large, qui finalisent la vocalisation par des voyelles d'appui et calculent les formes faibles.

Les structures de traits sont un formalisme très utilisé pour le traitement de la langue naturelle en général et la morphologie en particulier. Elles permettent de représenter de l'information partielle de façon économique et lisible. Karamel a la particularité de proposer une compilation statique en machines finies. Cela avait déjà été fait, (Kiraz, 1997 ; Zajac, 1998), mais à notre connaissance, aucune étude de cas n'a été publiée démontrant que ces techniques peuvent être utilisées pour des problèmes de taille réelle. La grammaire utilise sept types de structures de traits comportant treize traits ayant des domaines de valeurs de cardinalité comprises entre trois et quatorze. Des unifications avec variables sont utilisées pour décrire la circonfixation des formes conjuguées à préfixe, ce qui est un cas de dépendance à longue distance.

6. Comparaison avec d'autres travaux

Dans cette section nous allons comparer la grammaire du verbe akkadien avec divers travaux antérieurs concernant la description de la morphologie de langues sémitiques au moyen de machines finies à états. Parmi de très nombreux exemples, nous en avons sélectionné quatre représentatifs des grandes avancées et des différentes approches.

Ces travaux sont fortement influencés par un modèle théorique de la morphologie sémitique proposé par McCarthy : l'analyse *multiétages* (McCarthy, 1981). Dans ce modèle une forme de surface provient de la rencontre de quatre constituants : une racine, un schéma vocalique, un préfixe et un *patron* notant les successions de consonnes et voyelles. Voici un exemple tiré de (Beesley, 1998b) :

Étage du préfixe	n					
Étage de la racine		ʃ		b		r
Étage du patron	C	C	V	C	V	C
Étage de la vocalisation			a		i	
Radical	n	ʃ	a	b	i	r

L'opération consistant à *mélanger* différents éléments pour former un constituant est parfois appelée *interdigitation*. Il est possible de représenter chaque étage par une expression régulière. Les discontinuités sont représentées par la chaîne libre Σ^* et les éléments `C` et `V` du patron sont remplacés par la disjonction des consonnes et voyelles respectivement. Le radical peut être obtenu en réalisant l'intersection des différentes expressions régulières (Beesley, 1998b).

Commençons par le travail précurseur de Martin Kay. Pour traiter la partie non concaténative de la morphologie de l'arabe, il propose un transducteur multiruban traitant pratiquement directement une représentation multiétage à la McCarthy (Kay, 1987). La synchronisation n'est pas présente dans la représentation elle-même, mais réalisée au moyen d'une exécution non standard du transducteur. Il n'y a pas beaucoup de points communs entre cette approche et la grammaire de l'akkadien, si ce n'est l'usage de transducteurs multirubans.

La première grammaire décrivant une morphologie sémitique avec une machine finie a été une grammaire de l'akkadien, écrite en 1988, en morphologie à deux niveaux (Kataja et Koskenniemi, 1988). Il s'agit de la première démonstration de l'aptitude de ce modèle à décrire une morphologie non concaténative. Cette démonstration est partielle, dans la mesure où l'interdigitation ne pouvait pas être décrite avec le formalisme utilisé, celui du système `Kimmo`. Elle était réalisée au moyen de l'intersection d'un lexique de racines et d'un lexique d'éléments flexionnels. Cette opération n'étant pas présente dans le système utilisé, elle était effectuée comme un prétraitement fournissant le lexique des noyaux. Ensuite une description concaténative classique reliait les noyaux aux affixes et un système de règles à deux niveaux décrivait les alternances phonologiques affectant les formes lexicales.

La grammaire de Kataja et Koskenniemi paraît plus complète que la nôtre, puisqu'elle traite l'ensemble de la langue et pas seulement le verbe. Elle traite également davantage de phénomènes de surface du type assimilation. Notre description, en revanche, est plus fine pour ce qui est de la structure du noyau. La représentation utilisant une structure de traits est plus abstraite et correspond davantage à ce que l'on peut attendre d'un analyseur morphologique que la forme lexicale d'une grammaire à deux niveaux. La description en Karamel est plus lisible que celle en Kimmo. Le formalisme souffre moins de risque de conflits.

La morphologie à réécriture s'est aussi intéressée aux langues sémitiques, avec des apports importants. Une description morphologique de l'arabe a été réalisée chez

Xerox en utilisant `XFST`. Elle fait l'objet d'une démonstration en ligne[1], ainsi que de diverses publications (Beesley, 1998a ; Karttunen et Beesley, 2000).

La description structurelle fait intervenir deux éléments réunis par interdigitation : la racine et un motif comprenant les voyelles et les positions des consonnes de la racine. Par rapport aux étages de McCarthy, le motif est la contraction des étages de préfixe, de patron et de vocalisation : tous les éléments sont réalisés sauf les radicales dont la position est néanmoins marquée. Un exemple donné dans (Beesley, 2001) est le suivant : `[ktb&CaCaC]+Verb+FormI+Perf+Act` où le symbole `&` indique une interdigitation et les crochets délimitent la portée de cette opération. Une telle chaîne au niveau lexical est mise en correspondance au niveau de surface avec `katab`. Cet élément ternaire (racine, motif, traits) se comporte comme un affixe concaténé avec d'autres affixes qui ne comportent que deux éléments : une représentation lexicale de type habituel et des traits morphologiques. Un algorithme spécifique appelé *compile-remplace* a été développé pour réaliser l'interdigitation plus efficacement qu'avec l'algorithme d'intersection (Karttunen et Beesley, 2000).

Le niveau lexical de cette description est hétérogène. Les affixes ordinaires concatènent une représentation lexicale semblable à celle de la morphologie à deux niveaux et des traits. Le noyau, comme nous venons de le voir, concatène trois informations orthogonales. En Karamel, les informations de natures différentes sont mises sur des rubans différents, ce qui est plus satisfaisant. De plus, Karamel offre des structures de traits alors que XFST n'offre que des traits isolés dont la portée, l'affectation et l'unification doivent être spécifiées explicitement dans la grammaire. Karamel est donc plus déclaratif et plus convivial. La contrepartie se paie en termes d'efficacité. Les structures de traits peuvent s'avérer coûteuses et la gestion de multiples rubans présente également un léger surcoût. D'une certaine façon, une grammaire XFST ressemble à la version compilée d'une grammaire Karamel dans laquelle les différents rubans sont concaténés grain par grain et les traits sont représentés par des symboles. XFST est plus efficace et plus compact parce que l'écriture directe de la forme compilée ouvre la voie à certaines optimisations. Karamel n'utilise pas encore l'algorithme `compile-remplace` qui est plus efficace que l'intersection, mais rien n'empêcherait de l'intégrer pour optimiser les performances.

Une partie importante de la morphologie à partition a été développée spécifiquement pour la morphologie sémitique, et plus précisément le traitement de la langue syriaque par George Anton Kiraz (Grimley-Evans *et al.*, 1996 ; Kiraz, 2001). Ce modèle implémente l'analyse à plusieurs étages au moyen de transducteurs à plusieurs rubans. Dans la grammaire du syriaque, il y a trois étages (racine, vocalisation et patron comportant les préfixes) qui déterminent une forme de surface, ce qui fait un total de quatre rubans. Les règles contextuelles, inspirées de la morphologie à deux niveaux, offrent un centre qui correspond à peu près à notre notion de grain, mais les contextes n'utilisent pas cette notion. Ils décrivent séparément chaque ruban. Dans certains cas c'est assez pratique, mais il arrive que ce soit une limitation. On ne peut pas spécifier

1. URL : http ://www.xrce.xerox.com/competencies/content-analysis/arab

dans le contexte une correspondance entre des éléments de deux rubans. Par ailleurs, il n'y a qu'un seul type de grain et pas de possibilité d'imbrication. La mise en correspondance des chaînes de longueurs variables semble utilisée avec modération dans la grammaire. Cela concerne la possibilité de mettre en correspondance un élément d'un niveau avec la chaîne vide, et un élément de la racine avec ses deux occurrences dans les cas de gémination.

Une des spécificités de notre travail, par rapport aux quatre travaux que nous venons de présenter, est qu'il n'y a pas d'interdigitation, c'est-à-dire d'entrelacement de deux entités à constituants discontigus : il n'y a qu'un seul élément discontigu, à savoir la racine. Toutes les modifications de la racine se font par insertion d'un élément contigu (préfixe, infixe, géminée ou voyelle). Cela est lié aux particularités de l'akkadien et ne peut probablement pas être adapté à d'autres langues. Nous ne proposons pas une description à plusieurs étages de la langue. Ceci n'est pas dû aux contraintes du formalisme utilisé qui est, au contraire, tout à fait adapté à la multiplication des rubans. Le mécanisme d'abréviation (ou macro), dont nous avons montré l'utilisation pour le lexique de racines, permet d'avoir une description séparée des différents étages sans que cela se traduise par un ruban spécifique de la relation.

Notre grammaire a été développée après celles que nous avons mentionnées dans cette section. Elle a bénéficié de leurs apports. De la morphologie à deux niveaux vient la philosophie de la description morphotactique au moyen d'une intersection de contraintes locales. De la réécriture vient la cascade réalisant les alternances produisant la forme de surface. De la morphologie à partition viennent la multiplicité des niveaux et l'idée d'unité d'analyse (grain).

7. Conclusion

Dans cet article nous avons présenté un fragment de traitement automatique d'une langue dont les documents sont parmi les plus anciens écrits de toute l'humanité, une langue morte depuis plus de deux mille ans. Pour ce traitement, nous utilisons des techniques récentes, qui appartiennent encore au domaine de la recherche.

Le caractère mort de la langue a eu peu d'incidences sur notre travail. Il a comme conséquence que la phonologie de la langue reste hypothétique. C'est un point important quand il s'agit des alternances. Notre travail a été peu affecté par cette réalité car il ne comporte pas une description très fine des alternances. Il ne va pas au-delà des phénomènes principaux tels qu'ils sont décrits dans les ouvrages de référence.

Nos points de comparaison ne sont pas les travaux concernant les autres langues mortes ou anciennes, mais des langues linguistiquement apparentées, comme l'arabe, l'hébreu et le syriaque.

Notre travail peut se poursuivre dans sa direction actuelle pour compléter sa grammaire sur certains points inachevés : les clitiques, les verbes bi et quadrilitères, les verbes doublement faibles, les noms, les adjectifs. Améliorer la qualité de la des-

cription nécessiterait la collaboration d'experts de la langue à même d'apporter des éléments de connaissance diachronique et prosodique qui nous échappent.

8. Bibliographie

Barthélemy F., « Finite-State Compilation of Feature Structures for Two-Level Morphology », *International Workshop on Finite State Methods in Natural Language Processing (FSMNLP)*, Potsdam, Germany, 2007a.

Barthélemy F., « Multi-grain Relations », *Implementation and Application of Automata, 12th International Conference (CIAA)*, Prague, Czech Republic, p. 243-252, 2007b.

Barthélemy F., « Using Mazurkiewicz Trace Languages for Partition-Based Morphology », *ACL*, Prague (Czech Republic), 2007c.

Beesley K., « Finite-state morphological analysis and generation of Arabic at Xerox Research : Status and plans in 2001 », *ACL Workshop on Arabic Language Processing*, 2001.

Beesley K. R., « Arabic morphology using only finite-state operations », *in* M. Rosner (ed.), *Proceedings of the Workshop on Computational Approaches to Semitic languages*, 1998a.

Beesley K. R., « Arabic morphology using only finite-state operations », *Proceedings of the ACL Workshop on Computational Approaches to Semitic languages*, p. 50-57, 1998b.

Beesley K. R., Karttunen L., *Finite State Morphology*, CSLI Publications, 2003.

Breckwoldt T., Cunningham G., Black J., George A., Postgate N., *A Concise Dictionary of Akkadian*, Harrassowitz Verlag, 2000.

Buccellati G., *A Structural Grammar of Babylonian*, Harrassowitz Verlag, 1996.

Grimley-Evans E., Kiraz G., Pulman S., « Compiling a Partition-Based Two-Level Formalism », *COLING*, Copenhagen, Denmark, p. 454-459, 1996.

Heise J., « The Akkadian language », , `http ://www.sron.nl/~jheise/akkadian/`, 1995.

Huehnergard J., *A Grammar of Akkadian*, Eisenbrauns, Harvard Semitic Studies, 2005.

Kaplan R. M., Kay M., « Regular Models of Phonological Rule Systems », *Computational Linguistics*, vol. 20 :3, p. 331-378, 1994.

Karttunen L., « Constructing Lexical Transducers », *COLING-94*, Kyoto, Japan, p. 406-411, 1994.

Karttunen L., Beesley K. R., « Finite-State Non-Concatenative Morphotactics », *Fifth Workshop of the ACL Special Interest Group in Computational Phonology*, Luxembourg (Luxembourg), p. 1-12, 2000.

Kataja L., Koskenniemi K., « Finite-state description of semitic morphology : a case study of Ancient Akkadian », *Proceedings of the 12th conference on Computational linguistics*, Association for Computational Linguistics, Morristown, NJ, USA, p. 313-315, 1988.

Kay M., « Nonconcatenative finite-state morphology », *ACL Proceedings, Third European Conference*, p. 2-10, 1987.

Kiraz G. A., « Compiling Regular Formalisms with Rule Features into Finite-State Automata », *ACL*, Madrid, Spain, 1997.

Kiraz G. A., *Computational Nonlinear Morphology*, Cambridge University Press, 2001.

Koskenniemi K., « Two-Level Model for Morphological Analysis », *IJCAI-83*, Karlsruhe, Germany, p. 683-685, 1983.

Malbran-Labat F., *Manuel de langue akkadienne*, Publications de l'institut Orientaliste de Louvain (50), Peeters, 2001.

McCarthy J. J., « A Prosodic Theory of Nonconcatenative Morphology », *Linguistic Inquiry*, vol. 12, p. 373-418, 1981.

Sanchez R., « Cours d'akkadien », , disponible sur `http ://gedomia.ens-lsh.fr/`, 2005.

Wintner S., « Strengths and weaknesses of finite-state technology : A case study in morphological grammar development », *Nat. Lang. Eng.*, vol. 14, n° 4, p. 457-469, 2008.

Yli-Jyrä A. M., Koskenniemi K., « Compiling contextual restrictions on strings into finite-state automata », *Proceedings of the Eindhoven FASTAR Days 2004 (September 3–4)*, Eindhoven, The Netherlands, December, 2004.

Zajac R., « Feature Structures, Unification and Finite-State Transducers », *In FSMNLP'98 : International Workshop, on Finite State Methods in Natural Language Processing.*, 1998.