

L'utilisation des ordinateurs électroniques dans les sciences de l'homme

par ANTONIO A. M. QUERIDO

RÉSUMÉ

After a short introduction to the history of computers (in French *ordinateurs*) it is shown that they are not, exclusively, arithmetic machines: they can be used to manipulate qualitative as well as quantitative information.

Some types of qualitative information processing in the field of anthropological sciences are described: syntactic analysis, genealogical data ordination, and content analysis.

The communication of the researcher with the computer offers new difficulties in the field of anthropological sciences. Recently designed languages like SLIP (Symmetric List Processor) make it easy to translate into machine-readable form programs of structural ordination.

La recherche d'une technique d'ordination des symboles commence assez tôt dans l'histoire des sciences : on la trouve associée à la recherche d'un langage universel, pouvant servir dans les sciences de l'homme et dans celles de la nature ¹.

Cette recherche n'est jamais restreinte à l'ordination des symboles quantitatifs; si on remonte seulement au Moyen-Age, on remarque l'existence d'un ordinateur rudimentaire, celui de Raymond de Lulle, utilisé dans le domaine de l'information qualitative.

Le *Doctor Illuminatus* nous a aussi laissé un art de l'ordination, qu'il nomma *Ars Magna*, dans un livre du même titre publié en 1274.

L'idée fondamentale de l'*Ars Magna* était que chaque domaine de la connaissance peut être réduit à un ensemble de catégories élémentaires, et que la recherche philosophique n'est que l'exploration systématique de tous les arrangements possibles de ces catégories.

¹ Ce texte est une version modifiée d'une communication que l'auteur présenta au 33^e Congrès annuel de l'Association canadienne-française pour l'avancement des sciences, le 6 novembre 1966.

L'ordinateur de Lulle se composait de cercles concentriques de diamètres différents, capables d'effectuer des révolutions, indépendamment les uns des autres, autour d'un axe métallique. Sur le bord de chaque cercle, alignés sur des rayons équidistants, on gravait les symboles des catégories. Une rangée de symboles, lue du centre vers la périphérie, constituait un arrangement. La rotation d'un ou plusieurs cercles selon un *programme d'ordination*, engendrait des arrangements nouveaux.

L'art de Lulle a suscité des enthousiasmes et des controverses passionnées pendant des siècles. Parmi ses plus fervents admirateurs nous trouvons des hommes de valeur comme Giordano Bruno, qui a enseigné l'Ars Magna à l'aristocratie vénitienne, et Gottfried Wilhelm von Leibnitz. Celui-ci, dans son livre *Dissertatio de Arte Combinatoria* publié à Leipzig en 1666, voit dans l'art de Lulle le germe d'une *Mathesis Universalis* qui va permettre de ramener, un jour, la philosophie et les mathématiques à un seul système déductif. Quand il surgira des controverses, écrit Leibnitz, deux philosophes n'auront, alors, pas plus de raisons de se disputer que deux experts-comptables; il leur suffira de s'asseoir, de prendre leurs ardoises et de calculer.

Le complément logique d'une *Mathesis Universalis* serait un *ordinateur universel*. Mais le XVII^e siècle, marqué par les progrès dans le domaine des mathématiques quantitatives, s'intéressa surtout à l'ordination numérique. C'est ainsi que Blaise Pascal concevait, en 1642, sa machine à additionner et que Leibnitz lui-même inventait, en 1671, une machine capable aussi de multiplier.

On a tort de considérer ces calculatrices comme les ancêtres des ordinateurs électroniques. Elles n'ont pas introduit d'idées importantes pour le développement d'un ordinateur universel. Leurs vrais descendants sont les petites calculatrices de bureau.

Le premier ordinateur universel a été conçu vers le milieu du XIX^e siècle, par un professeur de Cambridge, Charles Babbage, et son associée Augusta Ada, comtesse de Lovelace, fille du poète Lord Byron.

Cet ordinateur, appelé *Analytic Engine*, comportait deux propriétés nouvelles, qui en faisait une machine extrêmement versatile : il avait une *mémoire* et il était *programmable*.

L'idée d'assembler les instructions nécessaires pour l'exécution d'une tâche en une *chaîne opératoire* et d'en équiper une machine n'était pas nouvelle. Jacquard avait déjà représenté, sur du carton perforé, des chaînes opératoires pour composer des motifs polychromes avec des machines à tisser.

L'innovation dans l'Analytic Engine, consiste à représenter la chaîne opératoire des ordinations par des symboles digitaux. La représentation symbolique d'une telle chaîne s'appelle un *programme*.

La communication symbolique avec le milieu extérieur entraîne, pour l'ordinateur, la nécessité d'une mémoire; en effet, il doit mémoriser et les symboles à ordonner et le programme d'ordination.

L'Analytic Engine n'a jamais été construit. Mais Lady Lovelace est allée jusqu'à écrire les premiers programmes. Elle a aussi décrit, avec beaucoup de pertinence, la nature de l'ordinateur programmable, c'est-à-dire l'ordinateur universel.

Les progrès de la technologie électronique ont finalement offert à l'homme les moyens appropriés pour la construction de l'ordinateur universel. Les premiers ordinateurs électroniques programmables sont EDVAC (Electronic Discrete Variable Automatic Calculator) de John von Neumann, aux États-Unis, et EDSAC (Electronic Delay Storage Automatic Calculator) de M. V. Milkes, en Angleterre. Ces deux ordinateurs ont été conçus vers 1945.

Avant de mémoriser un programme, l'ordinateur électronique est une machine indéfinie, une sorte de *tabula rasa*. Un programme de calcul numérique le transforme en une calculatrice alors qu'un programme d'analyse sémantique en fait une machine linguistique.

Puisque l'ordinateur mémorise le programme avant de commencer l'ordination, on peut dire qu'il possède une certaine autonomie; c'est ce qu'on veut dire quand on parle d'ordination *automatique*. Mais de là à attribuer à l'ordinateur une autonomie intégrale, il y a une distance qu'il ne faut pas sous-estimer. L'auteur du programme reste le vrai responsable des ordinations effectuées.

Actuellement, la plupart des universités disposent de leur propre ordinateur et cela nous semble tout naturel. Y a-t-il en effet, un cadre plus approprié pour l'ordinateur universel que *l'Universitas Studiorum* ?

Théoriquement, l'ordinateur universitaire est considéré comme un instrument de recherche mis à la disposition de tous les départements. En fait, il est utilisé surtout dans l'ordination numérique. Mais cet usage exclusif tend un peu partout à disparaître. Dans plusieurs universités, l'ordination des informations qualitatives est à la hausse. Les développements les plus importants se font dans le domaine de l'ordination linguistique. Il existe actuellement des centres universitaires d'ordination linguistique qui alimentent, à eux seuls, un gros ordinateur.

À l'Université de Montréal, qui vient d'être dotée d'un ordinateur de grande capacité, un groupe de professeurs et de chercheurs des sciences de l'homme est en train de développer un ensemble de programmes appropriés pour l'ordination d'informations qualitatives.

Le département de linguistique étudie l'organisation syntaxique de l'anglais en vue de tenter une traduction automatique de cette langue au français.

Les ordinations nécessaires pour traduire une langue dans une autre forment un long processus d'analyse-synthèse commençant, naturellement, par la reconnaissance des structures syntaxiques de la langue à traduire. La linguistique moderne a d'ailleurs établi que, pour arriver à un algorithme d'interprétation des phrases d'une langue, la reconnaissance des structures syntaxiques apparentes n'est pas suffisante. Il faut réduire ces structures apparentes aux structures latentes. Celles-ci constituent la base de l'interprétation sémantique des phrases, tandis que les structures apparentes ne fondent que leur représentation phonologique. Pour traduire ceci dans la terminologie de la glossématique, les structures latentes mettent la phrase en relation avec le plan du contenu et les structures apparentes, avec le plan de l'expression.

Les structures syntaxiques latentes sont mises en correspondance avec les structures apparentes par un ensemble de règles que l'on appelle *transformations*. Reconnaître les structures latentes d'une phrase équivaut, par conséquent, à renverser cette correspondance.

Ainsi, pour reconnaître les structures apparentes, le programme utilisera un dictionnaire, lequel indiquera, pour chaque mot, les éléments syntaxiques qu'il comporte. Ces éléments syntaxiques

sont ensuite assemblés en *arbres*, selon des règles grammaticales de contexte défini ou indéfini. Les arbres syntagmatiques, produits dans cette phase du programme, constituent les structures apparentes des phrases.

Les structures apparentes sont ensuite transformées pour obtenir les structures latentes. Ces transformations (*transformations renversées*) obéissent à des règles grammaticales quelque peu différentes des règles de catégorisation syntagmatique.

Une grammaire composée de règles de catégorisation syntagmatique et de règles de transformation se nomme *grammaire transformationnelle*.

Un autre programme que nous venons d'écrire, et qui intéresse surtout le département d'anthropologie, sert à établir des réseaux de parenté. Murdock fait une distinction entre relations de parenté *primaires* (celles exprimées par les mots *mère, père, fille, fils, sœur, frère, épouse, mari*), relations *secondaires* (celles qu'on peut exprimer avec deux de ces mots), relations *tertiaires*, etc. Notre programme reçoit, comme information initiale, l'arbre de relations primaires d'un individu et établit par la suite les relations secondaires, tertiaires, etc. Le degré d'expansion de l'*espace de parenté* est déterminé par l'anthropologue lui-même. Le programme doit être vérifié sur les structures de parenté d'une communauté esquimaude de Bathurst Inlet, Territoires du Nord-Ouest.

Le département de sociologie s'intéresse à un programme d'analyse thématique de textes ou d'enregistrements. L'ordination, dans ce cas, consiste à reconnaître et extraire d'un texte, segmenté et édité à l'avance, des thèmes élémentaires et des fonctions syntaxiques qui leur sont associées. Ce réseau de thèmes élémentaires et de fonctions permet — et c'est là la deuxième phase de l'ordination — de reconstituer des thèmes représentés par des syntagmes ou des phrases et de retrouver les segments du texte contenant ces thèmes.

Le programme d'analyse thématique s'applique à l'étude des informations codées en langage naturel, soit en sociologie, science politique, psychologie ou anthropologie.

On pourrait l'utiliser aussi pour classifier la documentation dans les sciences de l'homme et mettre sur pied un système de référence

automatique aux documents. Les textes classifiés seraient, dans ce cas, des résumés ou condensés de livres, d'articles scientifiques et de matériaux de recherche. L'index des thèmes, obtenu lors de la première phase de programmation, servirait au besoin de la référence pour répondre pertinemment à une demande bibliographique des utilisateurs.

Cet ensemble de programmes illustre bien la récurrence de certains problèmes dans l'ordination des informations qualitatives, quel que soit le domaine de la recherche envisagé. Dans chaque ensemble culturel étudié, qu'il s'agisse de syntaxe, de contenu sémantique ou de systèmes de parenté, la difficulté principale consiste à représenter dans la mémoire de l'ordinateur, chaque élément de cet ensemble replacé dans son contexte propre, ce qui permet de le saisir dans la plénitude de sa signification.

Les structures anthropologiques sont, en général, beaucoup plus complexes et beaucoup plus riches de signification que celles de l'ordination numérique.

C'est pourquoi, au Centre de calcul de l'université de Montréal, on a prévu l'utilisation de plusieurs langages de programmation, selon les divers types d'ordination à définir et à communiquer à l'ordinateur.

Pour l'ensemble des programmes que nous avons énumérés, on utilise deux langages de programmation. Les structures plus simples sont traduites en FORTRAN (Formula Translator) et les structures plus complexes, en SLIP (Symmetric List Processor), langage plus avancé permettant de mémoriser des arbres et des réseaux.

SLIP est un langage approprié à la programmation de l'ordination structurale. Avec ce langage, on réserve une partie de la mémoire de l'ordinateur dont les cellules sont soudées deux à deux, le module d'organisation de la mémoire opérant par paires de cellules. Chacune de ces paires devient un élément de SLIP.

La première cellule d'un élément sert à enchaîner les éléments d'une façon dynamique : un élément s'enchaîne à un autre s'il contient son adresse. En SLIP, l'enchaînement des éléments ne se fait pas en sens unique, c'est-à-dire que si l'élément A est relié à l'élément B, l'élément B est aussi relié à l'élément A. Les éléments d'un ensemble s'enchaînent donc de haut en bas et de bas en haut.

Ce mode d'enchaînement dynamique non seulement libère de la contiguïté physique dans la mémorisation d'un ensemble de symboles, mais aussi permet de réordonner ou transformer cet ensemble. On n'a pas besoin pour cela de déplacer les symboles; il suffit de changer les adresses qui enchaînent les éléments contenant ces symboles.

La première cellule d'un élément se divise en deux champs : dans le champ gauche, on trouve l'adresse de l'élément antérieur de la chaîne et, dans le champ droit, l'adresse de l'élément suivant. La deuxième cellule sert à mémoriser un symbole.

Les structures obtenues par ce système d'organisation dynamique de la mémoire se nomment *listes*. N'importe quel type de graphe peut être représenté sous forme de listes. Pour représenter une structure arborescente, par exemple, il suffira de traiter chaque sous-arbre comme une sous-liste et de placer l'adresse de cette sous-liste dans la deuxième cellule d'un élément d'une liste. Ceci revient à dire que les composants d'une liste sont, ou des symboles isolés, ou des listes de symboles. Les éléments d'embranchement des sous-listes dans les listes s'appellent *nœuds*.

La Figure 1 représente une liste de SLIP à trois niveaux de profondeur. Outre les éléments contenant des symboles à ordonner, nous pouvons voir les nœuds et, dans chaque liste ou sous-liste, l'élément *chef*.

L'élément chef d'une liste est le point d'entrée dans cette liste. Pour adresser l'ensemble de la liste, il suffit d'adresser son chef.

Comme pour les autres éléments d'une liste, la première cellule de l'élément chef contient les adresses d'enchaînement. Son adresse de gauche le relie au dernier élément de la liste précédente, son adresse de droite, au premier élément de la liste qui le suit.

La deuxième cellule de l'élément chef ne contient pas de symboles à ordonner, mais un *compteur de références* et l'adresse d'une liste d'attributs, s'il y a lieu. Le compteur de références indique le nombre des listes auxquelles celle-ci se rattache. La liste d'attributs sert à mémoriser des caractéristiques d'une liste donnée. En général, de deux éléments d'une liste d'attributs, le premier contient un attribut et le second, une valeur de cet attribut.

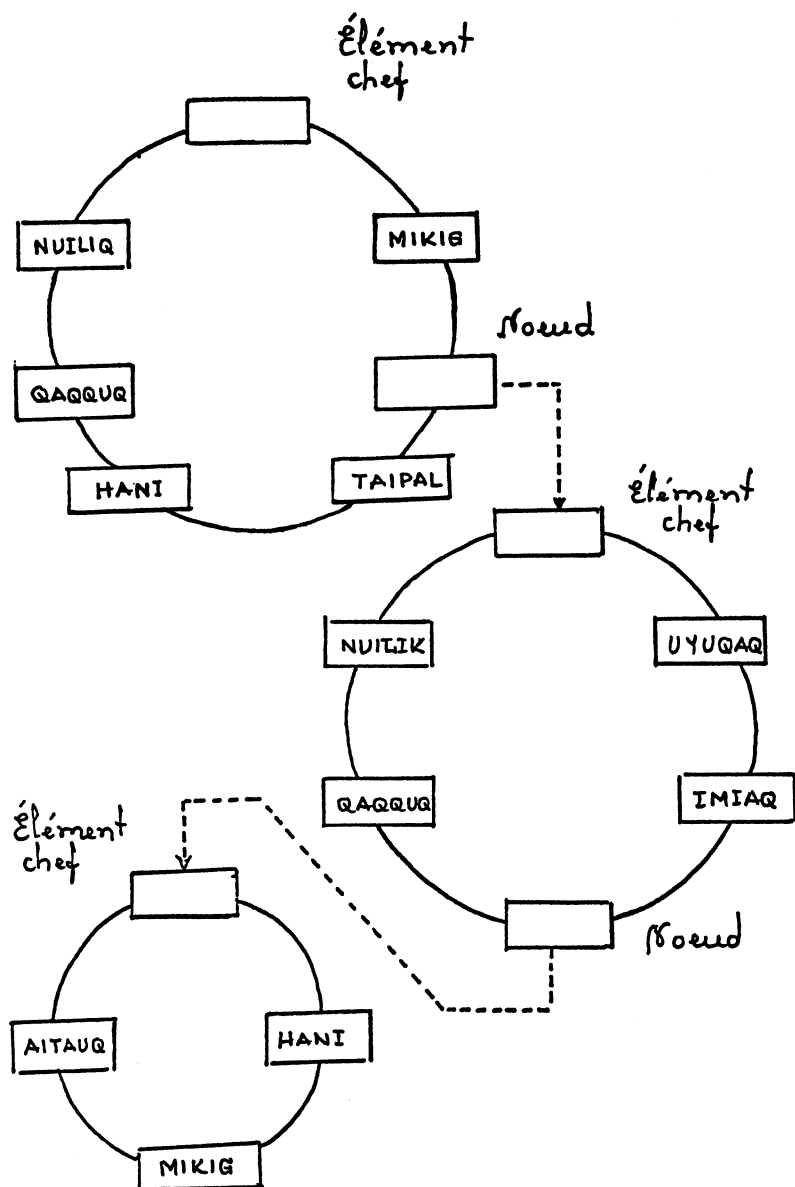


FIGURE 1

Ainsi, les listes permettent non seulement de représenter une structure dans toute sa complexité, mais aussi d'indiquer ses caractéristiques. Supposons qu'une sous-liste représente la structure d'un syntagme nominal d'une phrase : on peut associer à cette sous-liste une liste d'attributs contenant, par exemple, *fonction/sujet*. Ceci indiquerait que, dans la phrase, ce syntagme nominal fonctionne comme sujet. D'autres valeurs de l'attribut *fonction* pourraient être *objet*, *temps*, *lieu*, *mode*, *instrument*, etc.

Pour donner un exemple de structure représentable sous forme de listes, prenons un arbre de parenté esquimaude à plusieurs racines (cf. Figure 2). Chaque racine représente un individu, et ses branches connexes contiennent les noms de ses parents avec, pour chacun, le type de relation parentale impliquée : *father* (F), *mother* (M), *son* (S), *daughter* (D), *brother* (B), *sister* (Z), *husband* (H), ou *wife* (W). Chacun de ces sous-arbres constitue l'espace de parenté élémentaire d'un individu.

L'expansion de l'espace généalogique d'un individu peut se programmer en SLIP d'une façon extrêmement simple : il suffit d'insérer dans la liste de l'espace généalogique élémentaire les adresses des listes de l'espace généalogique élémentaire de chacun des parents. Les flèches dans la Figure 3 montrent ce processus d'insertion.

C'est de cette façon qu'on dote la mémoire de l'ordinateur d'un réseau de parenté extrêmement complexe. Pour chaque individu, le programme lit son espace généalogique complexe et l'imprime. Ainsi, la Figure 4 nous donne une partie des résultats obtenus pour un groupe d'Esquimaux de Bathurst Inlet, T.N.O.²

L'ordination par listes peut s'appliquer à d'autres types de structures anthropologiques. Il est, par exemple, très approprié de représenter en SLIP les structures syntaxiques, sémantiques et phonologiques d'un langage naturel.

SLIP est un des langages d'ordination structurale les plus puissants qui existent actuellement. Se prêtant bien au traitement de l'information qualitative de structure complexe, il peut être un

² Ces données ont été recueillies par M. Rémi Savard dans sa mission d'été 1965.

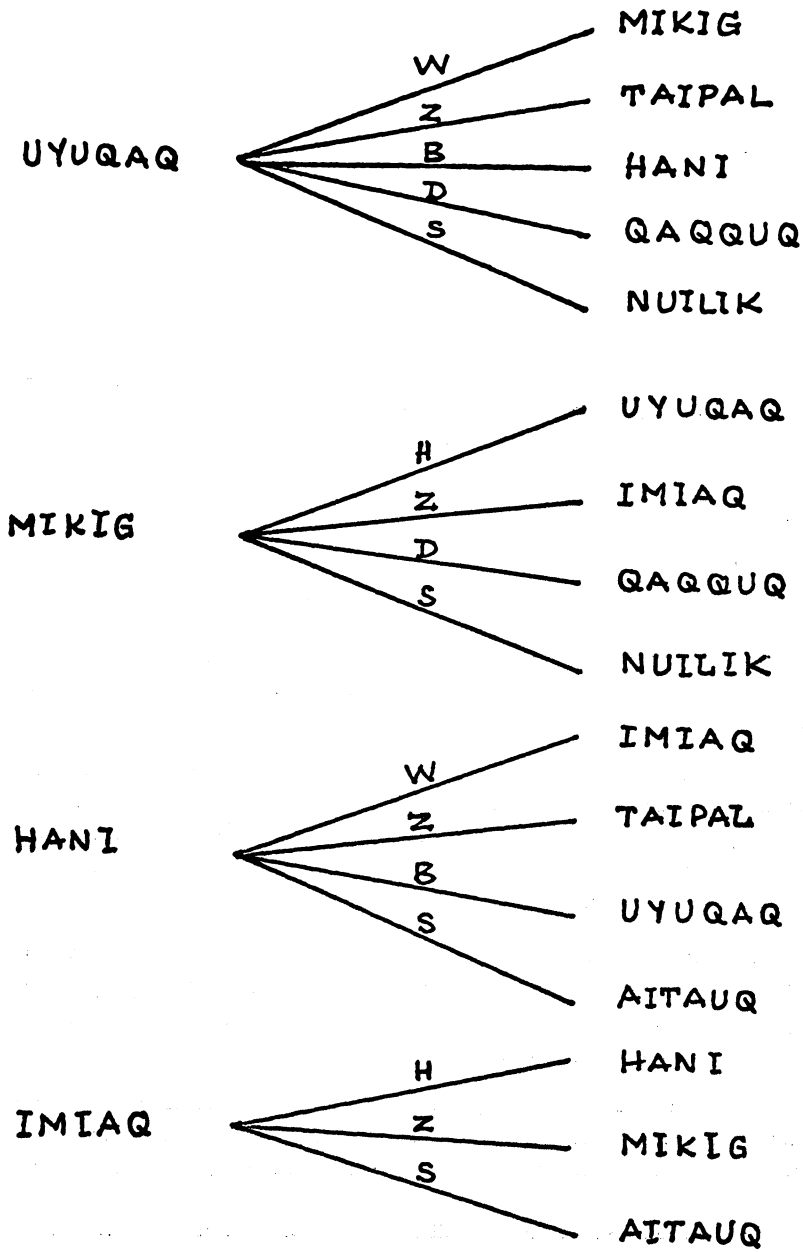


FIGURE 2

instrument de recherche très utile pour les anthropologues. Il leur permettra de procéder à de nouvelles recherches dans le domaine de l'analyse structurale et de communiquer directement avec l'ordinateur.

Un autre avantage de SLIP est sa filiation avec un des langages de programmation scientifique les plus répandus, le système FORTRAN.

SLIP peut être considéré comme un dialecte (ou, plus précisément, un *jargon*) de FORTRAN. N'importe quel connaisseur du français, par exemple, peut reconnaître la structure des phrases d'un article scientifique écrit en français. Il peut aussi interpréter un certain nombre de mots autres que ceux du jargon de la science en question. Les mots de jargon font donc toute la différence entre le français commun et le français des articles scientifiques, la structure syntaxique étant identique dans les deux cas. De même, les phrases de SLIP sont sujettes à la grammaire de FORTRAN. Tout programmeur de FORTRAN comprend, structurellement, une phrase en SLIP. Mais, très souvent, il ne peut pas l'interpréter complètement parce qu'il ignore ce que font les sous-programmes dont le nom apparaît dans les phrases de SLIP.

Apprendre SLIP, quand on connaît FORTRAN, c'est donc apprendre à utiliser un ensemble de sous-programmes enregistrés sur bande magnétique, et à les combiner avec art pour ordonner des informations qualitatives hautement structurées.

FORTRAN étant un langage approprié à l'ordination des informations quantitatives, le système de programmation FORTRAN — SLIP se prête à une application fort étendue.

Les sciences de l'homme, utilisant à la fois des données quantifiables et des données qualitatives, offrent un éventail d'ordinations idéal à l'utilisation de ce système.

Département de linguistique et
Centre de calcul,
Université de Montréal.

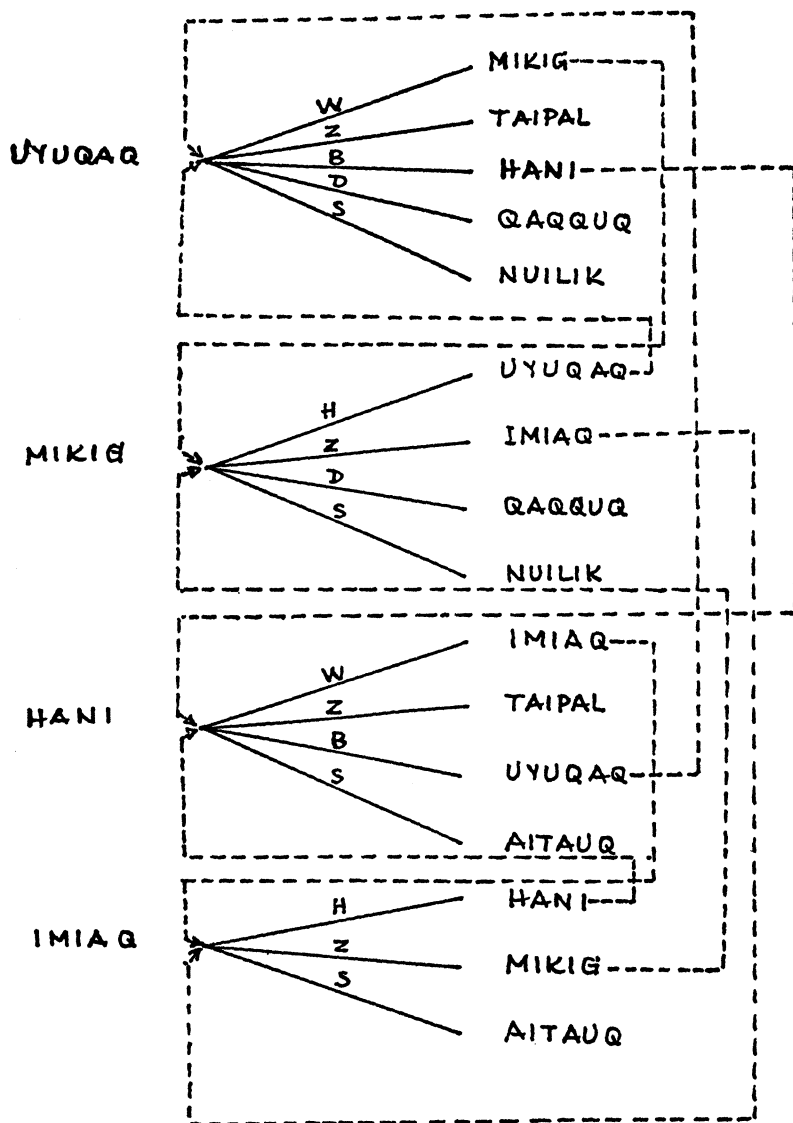


FIGURE 3

RELATIVES OF MIKIG1

HUSBAND'S SISTER'S SON'S WIFE'S BROTHER'S SON'S WIFE'S FATHER	AITAUQ3
HUSBAND'S BROTHER'S WIFE'S SISTER	HAN15
HUSBAND'S BROTHER'S WIFE'S SISTER	IMIAQ1
HUSBAND'S BROTHER'S WIFE'S SISTER	MIKIG1
HUSBAND'S BROTHER'S SON	AITAUQ3
HUSBAND'S BROTHER'S SON'S WIFE	PAGLIK1
HUSBAND'S BROTHER'S SON'S DAUGHTER	AOUNGAR2
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND	UYUQAQ2
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER	OOMOYA3
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER	TAIPAL
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S BROTHER	UYUQAQ1
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S FATHER	HAN15
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S WIFE	KIVAC3
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S WIFE	IMUYAQ
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER	QAQQUQ1
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S MOTHER	NULIK2
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S FATHER	MIKIG1
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER	UYUQAQ1
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER	QAQQUQ1
HUSBAND'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER'S HUSBAND	IMUYAQ
SISTER	IMIAQ1
SISTER'S HUSBAND	HAN15
SISTER'S HUSBAND'S SISTER	TAIPAL
SISTER'S HUSBAND'S SISTER'S HUSBAND	KIVAC3
SISTER'S HUSBAND'S SISTER'S DAUGHTER	OOMOYA3
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND	NULIK2
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND'S MOTHER	MIKIG1
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND'S FATHER	UYUQAQ1
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND'S SISTER	QAQQUQ1
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND'S SISTER'S HUSBAND	IMUYAQ
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND'S SISTER'S HUSBAND'S MOTHER	TAIPAL
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND'S SISTER'S HUSBAND'S FATHER	KIVAC3
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S HUSBAND'S SISTER'S HUSBAND'S SISTER	OOMOYA3
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S SON	UYUQAQ2
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S SON'S WIFE	AOUNGAR2
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S SON'S WIFE'S MOTHER	PAGLIK1
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S SON'S WIFE'S FATHER	AITAUQ3
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S SON'S WIFE'S FATHER'S MOTHER	IMIAQ1
SISTER'S HUSBAND'S SISTER'S DAUGHTER'S SON'S WIFE'S FATHER'S FATHER	HAN15
SISTER'S HUSBAND'S SISTER'S SON	IMUYAQ

FIGURE 4 (suite)

RELATIVES OF MIKIG1

SISTER'S HUSBAND'S SISTER'S SON'S WIFE'	QAQOQ1
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S MOTHER	MIKIG1
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S FATHER	UYUQAQ1
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S BROTHER	NULIK2
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S BROTHER'S WIFE	OOMOYA3
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S BROTHER'S WIFE'S MOTHER	TAIPAL
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S BROTHER'S WIFE'S FATHER	KIVAG3
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S BROTHER'S WIFE'S BROTHER	IMUYAO
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S BROTHER'S SON	UYUQAQ2
SISTER'S HUSBAND'S SISTER'S SON'S WIFE'S BROTHER'S SON'S WIFE	AOUNGAR2
SISTER'S HUSBAND'S BROTHER	UYUQAQ1
SISTER'S SON	AITAUQ3
SISTER'S SON'S WIFE	PAGLIK1
SISTER'S SON'S DAUGHTER	AOUNGAR2
SISTER'S SON'S DAUGHTER'S HUSBAND	UYUQAQ2
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER	OOMOYA3
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER	TAIPAL
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S BROTHER	UYUQAQ1
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S BROTHER'S WIFE	HAN5
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S BROTHER'S SON	IMIAQ1
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S BROTHER'S SON'S WIFE	AITAUQ3
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S BROTHER'S SON'S WIFE	KIVAG3
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S WIFE	IMUYAO
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S WIFE'S MOTHER	QAQOQ1
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S WIFE'S FATHER	MIKIG1
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S WIFE'S BROTHER	UYUQAQ1
SISTER'S SON'S DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S WIFE'S BROTHER'S BROTHER	NULIK2
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S MOTHER	NULIK2
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S MOTHER'S FATHER	MIKIG1
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S FATHER	UYUQAQ1
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER	QAQOQ1
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER'S HUSBAND	IMUYAO
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER'S HUSBAND'S MOTHER	TAIPAL
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER'S HUSBAND'S FATHER	KIVAG3
SISTER'S SON'S DAUGHTER'S HUSBAND'S FATHER'S SISTER'S HUSBAND'S SISTER	OOMOYA3
DAUGHTER	QAQOQ1
DAUGHTER'S HUSBAND	IMUYAO
DAUGHTER'S HUSBAND'S MOTHER	TAIPAL
DAUGHTER'S HUSBAND'S MOTHER'S BROTHER	UYUQAQ1
DAUGHTER'S HUSBAND'S MOTHER'S BROTHER'S BROTHER	HAN5

FIGURE 4 (suite)

RELATIVES OF MIKIG1

DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S WIFE	IMIAQ1
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S WIFE'S SISTER	MIKIG1
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S SON	AITAUQ3
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S SON'S WIFE	PAGLIK1
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S SON'S DAUGHTER	AOUNGAR2
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S SON'S DAUGHTER'S HUSBAND	UYUQAQ2
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER	OOMOYA3
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER	NUILIK2
DAUGHTER'S HUSBAND'S	MOTHER'S BROTHER'S FATHER	KIVAG3
DAUGHTER'S HUSBAND'S	SISTER	OOMOYA3
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S	NUILIK2
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S MOTHER	MIKIG1
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S FATHER	UYUQAQ1
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER	OAOOUQ1
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S HUSBAND	IMIAQ1
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S HUSBAND'S MOTHER	UYUQAQ1
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S HUSBAND'S MOTHER'S BROTHER	HAN15
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S HUSBAND'S FATHER	KIVAG3
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S HUSBAND'S SISTER	OOMOYA3
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S HUSBAND'S SISTER'S HUSBAND	NUILIK2
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S HUSBAND'S SISTER'S HUSBAND'S SON	UYUQAQ2
DAUGHTER'S HUSBAND'S	SISTER'S HUSBAND'S SISTER'S WIFE	UYUQAQ2
DAUGHTER'S HUSBAND'S	SISTER'S SON	AOUNGAR2
DAUGHTER'S HUSBAND'S	SISTER'S SON'S WIFE'S MOTHER	PAGLIK1
DAUGHTER'S HUSBAND'S	SISTER'S SON'S WIFE'S FATHER	AITAUQ3
DAUGHTER'S HUSBAND'S	SISTER'S SON'S WIFE'S FATHER'S MOTHER	IMIAQ1
DAUGHTER'S HUSBAND'S	SISTER'S SON'S WIFE'S FATHER'S MOTHER'S SISTER	MIKIG1
DAUGHTER'S HUSBAND'S	SISTER'S SON'S WIFE'S FATHER'S FATHER	HAN15
DAUGHTER'S HUSBAND'S	SISTER'S SON'S WIFE'S FATHER'S SISTER	TAPAL
DAUGHTER'S HUSBAND'S	SISTER'S SON'S WIFE'S FATHER'S FATHER'S BROTHER	UYUQAQ1
SON	NUILIK2	NUILIK2
SON'S WIFE	OOMOYA3	OOMOYA3
SON'S WIFE'S MOTHER	TAPAL	TAPAL
SON'S WIFE'S MOTHER'S BROTHER	UYUQAQ1	UYUQAQ1
SON'S WIFE'S MOTHER'S BROTHER'S WIFE	HAN15	HAN15
SON'S WIFE'S MOTHER'S BROTHER'S WIFE'S WIFE	IMIAQ1	IMIAQ1
SON'S WIFE'S MOTHER'S BROTHER'S WIFE'S SISTER	MIKIG1	MIKIG1
SON'S WIFE'S MOTHER'S BROTHER'S SON	AITAUQ3	AITAUQ3
SON'S WIFE'S MOTHER'S BROTHER'S SON'S WIFE	PAGLIK1	PAGLIK1

FIGURE 4 (suite)

RELATIVES OF MIKIGI	
SON'S WIFE'S MOTHER'S BROTHER'S SON'S DAUGHTER	AOUNGAR2
SON'S WIFE'S MOTHER'S BROTHER'S SON'S DAUGHTER'S HUSBAND	UYUQAQ2
SON'S WIFE'S MOTHER'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S MOTHER	OOMOYA3
SON'S WIFE'S MOTHER'S BROTHER'S SON'S DAUGHTER'S HUSBAND'S FATHER	NUILK2
SON'S WIFE'S FATHER	KIVAG3
SON'S WIFE'S BROTHER	IMUYAO
SON'S WIFE'S BROTHER'S WIFE	OAOQUO1
SON'S WIFE'S BROTHER'S WIFE'S MOTHER	MIKIG1
SON'S WIFE'S BROTHER'S WIFE'S FATHER	UYUQAQ1
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S	NUILK2
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE	OOMOYA3
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE'S MOTHER	TAIPAL
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE'S MOTHER'S BROTHER	UYUQAQ1
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE'S MOTHER'S BROTHER	HANI5
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE'S FATHER	KIVAG3
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE'S BROTHER	IMUYAO
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE'S BROTHER'S WIFE	OAOQUO1
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S SON	UYUQAQ2
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S SON'S WIFE	AOUNGAR2
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S SON'S WIFE'S MOTHER	PAGLIK1
SON'S WIFE'S BROTHER'S WIFE'S BROTHER'S SON'S WIFE'S FATHER	AITAUQ3
SON'S SON	IMIAQ1
SON'S SON'S WIFE	MIKIG1
SON'S SON'S WIFE'S MOTHER	HANI5
SON'S SON'S WIFE'S FATHER	TAIPAL
SON'S SON'S WIFE'S FATHER'S	KIVAG3
SON'S SON'S WIFE'S FATHER'S FATHER'S	OOMOYA3
SON'S SON'S WIFE'S FATHER'S FATHER'S SISTER	NUILK2
SON'S SON'S WIFE'S FATHER'S FATHER'S SISTER'S DAUGHTER	UYUQAQ2
SON'S SON'S WIFE'S FATHER'S FATHER'S SISTER'S DAUGHTER'S HUSBAND	IMUYAO
SON'S SON'S WIFE'S FATHER'S FATHER'S SISTER'S DAUGHTER'S SON	OAOQUO1
SON'S SON'S WIFE'S FATHER'S FATHER'S SISTER'S SON'S WIFE	UYUQAQ1
SON'S SON'S WIFE'S FATHER'S FATHER'S SISTER'S SON'S WIFE'S BROTHER	

FIGURE 4 (suite)

REFERENCES

- BORKO, HAROLD, *ed.*
1962 Computer Applications in the Behavioral Sciences. Prentice-Hall.
- CARRERAS YARTAU, Tomas
1939 Historia de la filosofia espanola. Madrid.
- CHOMSKY, NOAM
1965 Aspects of the Theory of Syntax. Cambridge, M.I.T. Press.
- COULT, ALLAN D. and RICHARD R. RANDOLPH
1965 Computer Methods for Analyzing Genealogical Space. American Anthropologist 67.
- GARDNER, MARTIN
1958 Logic Machines and Diagrams. New York, McGraw-Hill.
- LEIBNITZ, GOTTFRIED W. VON
1666 Dissertatio de arte combinatoria. Leipzig.
- LULO, RAYMUNDO
1274 Ars Magna.
- MURDOCK, GEORGE P.
1949 Social Structure. New York, MacMillan.
-