# The kinship as a computational question

Carlos Eduardo Ferreira[1,2]

*Department of Computer Science, Institute of Mathematics and Statistics*
*University of São Paulo*

Álvaro Junio Pereira Franco[1,*]

*Department of Computing, Center of Science, Technology and Health*
*Federal University of Santa Catarina*

Marcio Ferreira da Silva[3,*]

*Department of Anthropology, Faculty of Philosophy, Language and Literature, and Human Science*
*University of São Paulo*

**Abstract**

We present some problems of the Structural Anthropology area and we discuss about their computational complexity. We use mixed graphs to model the problems. Our interest is in developing algorithms to enumerate structures that occur in determined kinship networks. These structures are called rings. The rings of some people contain attributes as the rings with nomination and formal friendship connections from the Krahô people; and the chromatic rings from the Enawenê-Nawê people where each individual has a color (the color from the group that the individual belongs). In these cases, the rings of our interest have to obey a kind of pattern on the arcs and colors on the vertices.

We also present a way to enumerate all rings of a kinship network without any particularity.

*Keywords:*
Algorithm, Computational Complexity, Kinship Networks

[*]Corresponding author.
[1]Supported by CNPq Proc. 456792/2014-7.
[2]Supported by FAPESP Proc. 2013/03447-6 and CNPq Proc. 305705/2014-8.
[3]Supported by ???

## 1. Introduction

A way to understand the social behavior of a people can be given through the study of the exchanges which occur among their individuals. The individual circulation between families creates ties among individuals that were strangers to each other before the exchange (and by hypothesis). However, once consolidated the exchange, the treatment among the members of these families becomes different. Now, the individuals are not strangers anymore. They are *relatives*.

The Anthropology area that study the kinship is an interesting area. The *parent relationships* such as *father*, *mother*, *son*, and *daughter*, and the *affinity relationships* such as *husband* and *wife*, are primitives that give us the opportunity to understand the social behavior of a people. It is very natural to represent the parent and affinity relationships using a combinatorial structure composed by a set of individuals (or vertices), and by sets of connections between individuals (arcs and edges without orientation) where:

- an arc from a vertex $u$ to a vertex $v$ represents a connection of consanguinity (or parent relationship), that is, $u$ is father (or mother) of $v$;
- an edge between vertices $u$ and $v$ represents a connection of affinity between $u$ and $v$, in other words, $u$ and $v$ are *married*.

The use of graphs in the representation of individuals and their connections has been used since (at least) Ore [1]. In Anthropology and Computation, such structure is called *kinship network* ([2] and [3]). The investigation of combinatorial structures in mixed graphs (i.e., graphs composed by a set of vertices, a set of arcs and a set of edges), analysis on kinship networks and the development and implementation of graph algorithms are important activities for who is interested in this area.

Consider a kinship network $G$. Some marriages of $G$ can be deduced even if there are no edges representing them. For example, we can say that individuals $u$ and $v$ are affinal when there are arcs from $u$ to $w$ and from $v$ to $w$, where $w$ is an other individual of the network. When we say that *marriages determine individuals*, it is natural since an individual is born by means of an affinity relationship. On the other hand, followers of the *Alliance Theory* believe that the opposite is also a valid argument, i.e., *individuals determine marriages*. There are cases in our own society that we can contrast, and can be used to confirm that hypothesis. The study of rules that determine how the people alliances occur, has its begin in the work of the anthropologist Lévi-Strauss [4]. In any society, there are *exchange rules* that could enable or could not enable matrimonial alliances among its individuals, and therefore, the circulation of their people and the social life of them, would be products of the rules that encourage or turn impossible the matrimonial alliances. In the book of Dumont [5], we can see an example of a society where a man who cannot have sexual contact with his close relatives such as his sister or mother (the incest forbidden), he looks for an other man to be husband of his close relatives, receiving as exchange, his wife.

Assuming that marriages are determined by individuals (not by some random effect), structures caused by *marriage rules* can be considered in the kinship study. For example, imagine the following rule: *it is good to a man to marry to the Mother's Daughter's Husband's Mother's Daughter*, MDHMD for short, (in other words, with the Sister's Husband's Sister, ZHZ for short). A certain structure on the network starts to appear. The Fig. 1 illustrates two occurrences MDHMD (or ZHZ). If the frequency of this pattern in a network is high, then it is possible to conclude that the corresponding society has a tendency to marry individuals following such pattern. In the literature these patterns are called *rings* or *matrimonial circuits* [6], [7], [3]. Note that a ring is the simplest structure capable to connect members of distinct families. That is the reason we have interest in these structures.

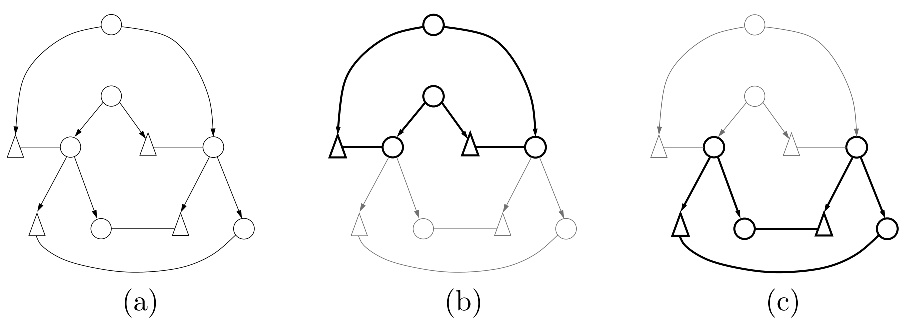


Figure 1: Triangles and circles represent male and female individuals, respectively. A part of a kinship network in (a). Two occurrences of a pattern are illustrated in (b) and (c).

The construction of tools able to find rings in kinship networks is an important contribution from the Computing to the Anthropology. However, to find rings in networks is a NP-hard problem as we will see in Section 5. Despite the difficulty of finding rings, anthropologists still need solutions for the problem. So, it is necessary to develop practical tools (and algorithms) to solve them. We believe that the followings activities are fundamentals to the area:

- development of tools to assist anthropologists in the analysis of kinship networks;
- correlation of Anthropological problems and Computing problems;
- study and understand of the difficulty to find rings in particular networks; and
- development and implement algorithms to find/enumerate rings.

It is important to emphasize the following. Methods to find combinatorial structures in kinship networks and methods to analyze such structures have been developed in the last years [8], [9], [10], [7], [11] and [12]. This article follows the same goal. In specific, we try to clarify the procedure developed for us to obtain rings in kinship networks. This procedure, in our point of view, it was fundamental to consolidate our tool in the enumeration of rings. We did some tests in 8 empirical kinship networks and we note that our procedure works experimentally very well. Still in this text, we present some anthropological problems and their computational complexity.

*Kinship Machine* (or Máquina do Parentesco in Portuguese, *MaqPar* for short) is the name of a tool that we have been working on. This tool was initially developed by Silva and Poz [7] and implemented in a query language to relational database. In the last years, we have tried to transform MaqPar in a tool faster and multiplatform.

The notation used in this text is described in Section 2. In Section 3, we describe a way to enumerate rings in kinship networks. It is implemented in our tool. Some preliminary results are presented in Section 4. In Sections 5, 6 and 7, we present some challenging problems that we have interest in solution produced by good algorithms. The conclusions are in Section 8.

## 2. Notation

Our notation is based on works from [7] and [3]. A kinship network is composed by a set of vertices (individuals), a set of arcs (consanguinity connections) and a set of edges (affinity connections). In terms of Graph Theory, a kinship network is a *mixed graph*.

An *oriented cycle* is a digraph where each vertex has in-degree and out-degree equal to 1. A *cycle* in a mixed graph is a subgraph where each vertex has degree equal to 2 (we are counting enter arcs, out arcs, and edges). There is no oriented cycle in kinship networks as it is not possible an individual be his/her own proper ancestor, however, there would be exist cycles. A *parental triangle* is a cycle induced by three vertices: an individual and his/her parents. The vertices and arcs of an parental triangle formed a *parental triad* (see Fig. 2). Finally, a *ring* is a cycle without parental triad.

(a) (b)

Figure 2: (a) A parental triangle and (b) a parental triad.

Actually, the rings found by MaqPar have at most three[4] affinity connections, and they are classify as the following:

- consanguinity marriages (or A1C1);
- consanguinity relinking marriages (or A2C1, A2C2); and
- affinal relinking marriages (or A3C1, A3C2, A3C3).

The notation A$k$C$l$ (with $k$ and $l$ being positive integer numbers, and $k \geq l$) denotes a ring with $k$ affinity connections and $l$ ancestors with disjoint consanguinity lines to individuals which are married (not necessary to each other). The highlight structures in Fig. 1 (b) and (c) are examples of A2C2 rings. In Fig. 3 are illustrated A1C1, A2C1 and A3C3 rings. In Fig. 3 (b), it is illustrated an A2C1 ring with 2 affinity connections ($u - v$ and $w - v$) and 1 common ancestor to $u$ and $w$ (namely $s_1$) with disjoint consanguinity lines to individuals $u$ and $w$. In Fig. 3 (c), it is illustrated an A3C3 ring with 3 affinity connections ($v - u$, $w - z$ and $x - y$) and 3 common ancestors of individuals ($s_1$, $s_2$ and $s_3$) with disjoint consanguinity lines to the individuals $u$ and $w$, $z$ and $y$, and $x$ and $v$.

In a ring, the common ancestors with disjoint consanguinity line are also called *junctions* [12]. In general terms, a vertex $s$ is a *junction* of a pair of vertices $\{u, v\}$, if there are internally vertex-disjoint directed paths from $s$ to $u$ and from $s$ to $v$.

[4] The tool can be easily adopted to find rings with any number of affinity connections.

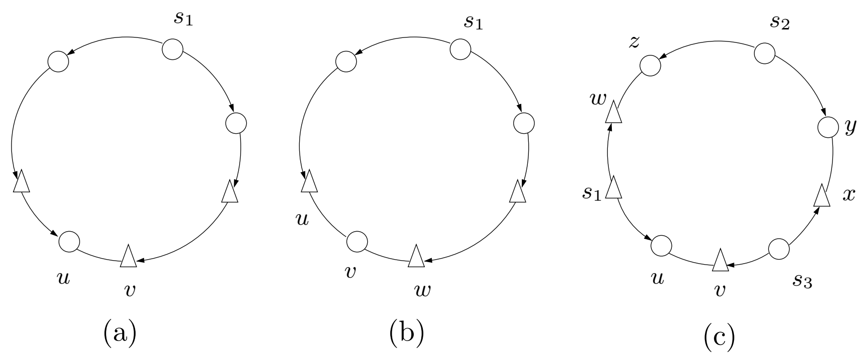


Figure 3: (a) A1C1, (b) A2C1, and (c) A3C3. The vertices with label $u, v, x, y, w$ and $z$ are in affinity connections. The vertices with label $s_i$ are common ancestors with disjoint consanguinity lines to married individuals.

Up to now, our main result was a fast algorithms to find junctions in acyclic digraphs [13], [14]. An digraph is acyclic if it has no oriented cycles. Note that it is the case of kinship networks when we look at to the digraph induced by the arcs of the network. In this way, we can use such algorithms to the cases treated here. In fact, they already were incorporated to MaqPar which actually enumerates all A$k$C$k$ rings for $k = 1, 2, 3$.

We finish this section describing a notation to directed paths (or dipaths) and rings in kinship networks. Let $D$ be a kinship network. Consider two individuals $u$ and $v$ in $D$. A dipath from $u$ to $v$ is denoted as a tuple $P = (u = w_0, w_1, \ldots, w_k = v)$ where $w_i \to w_{i+1}$ is an arc in $D$ for $i = 0, 1, \ldots, k-1$. The inverse dipath of $P$ is denoted by $\overleftarrow{P}$. The concatenation of two dipaths $P$ and $Q$ is denoted by $PQ$. In this case, if $P$ finishes with a determined vertex $w$, then $Q$ must begin with the same vertex $w$. We allow the concatenation $\overleftarrow{P}Q$ whether $P$ and $Q$ start with the same vertex.

The set of junctions of $u$ and $v$ is denoted by $J_{uv}$. Fixed a vertex $s$, we allways have $s$ in $J_{sv}$ if there is a dipath from $s$ to $v$. It means that $s$ is a junction of pairs $s$ and $v$ for which there is a dipath from $s$ to $v$. If there are internally vertex-disjoint dipaths from $s$ to $v$, then $s$ is in $J_{vv}$. Lastly, the set of all dipaths from $u$ to $v$ is denoted by $P_{uv}$. Considering Fig. 4 as an example, we can find the set of all junctions of vertices label in the figure as $u$ and $v$: $J_{uv} = \{s_1, s_3, s_8\}$. The sets formed by all dipaths from $s_1$ to $u$ and from $s_1$ to $v$ are, respectively, $P_{s_1u} = \{(s_1, s_3, s_6, s_8, u)\}$ and $P_{s_1v} = \{(s_1, s_3, s_6, s_8, v), (s_1, s_3, s_7, v), (s_1, s_4, s_7, v)\}$.

Note that the dipaths $P = (s_1, s_3, s_6, s_8, u)$ and $Q = (s_1, s_4, s_7, v)$ are internally vertex-disjoint, that is, the single common vertex to $P$ and $Q$

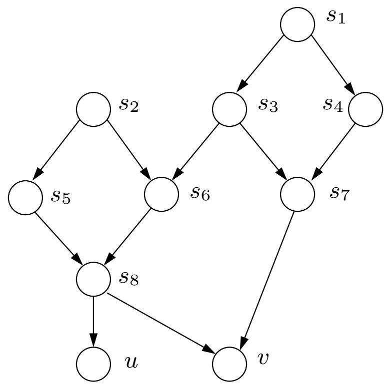


Figure 4: An hypothetical kinship network, with no edges (marriages) and no distinction between the gender of the individuals.

is $s_1$. This show that $s_1$ is a junction of vertices $u$ and $v$. In a similar way, we can show that the vertices $s_3$ and $s_8$ are also junctions of $u$ and $v$. Finally, note that, if vertices $u$ and $v$ are married to each other, then $\overleftarrow{P}Q$ is an A1C1 ring. In our example, the ring is denoted as the following $\overleftarrow{P}Q = (u, s_8, s_6, s_3, s_1, s_4, s_7, v)$. The representation of the marriage of $u$ and $v$ is implicit. The MaqPar considers such notation in the enumeration of the rings. Next section, we described in details, a way to obtain A$k$C$k$ rings of a network. It is implemented in MaqPar.

## 3. A way to enumerate all A*k*C*k* rings

In this section we describe a procedure that enumerates all A$k$C$k$ rings. It is currently implemented in our tool.

Throughout this section, consider a kinship network $D$, and a set of pair of vertices $F$ which is formed by

- all marriages $(u, v)$, when we are treating the case of finding all A1C1 rings; or
- all pairs of vertices $u$ and $v$, married but not to each other, when we are treating the case of finding all A$k$C$k$ ($k \geq 2$) rings.

Next we describe the three main steps of the procedure of finding all A$k$C$k$'s:

1. Find the sets of all junctions of all pairs of vertices in $F$;

2. Build an sorted set with $k$ fixed marriages $C = \{(u_1, v_1), \ldots, (u_k, v_k)\}$; and
3. Use a procedure which finds all rings such that $C$ (the $k$ fixed marriages in the previous step) occurs in the ordering given in $C$.

We describe bellow each cited step.

### *3.1. Step 1*

To find the sets of all junctions of pairs in $F$, we use an algorithm that, given a vertex $s$ in the network $D$, it partitions a set of vertices denoted by $V_s$ in such a way that *two vertices $u$ and $v$ are in different sets in the partition if and only if $s$ is a junction of $u$ and $v$.* $V_s$ is the set of vertices $v$ such that there exist a dipath from $s$ to $v$.

In each set of the partition we can have many vertices of $D$. However, each set will have a unique *representative*. We describe such algorithm in the following. Along the years, we were making its description better as in [13] and [14]. Nevertheless, we observe another way to describe such algorithm that we presented here. Up to now, it is its simplest description.

It is given a kinship network $D$ and a vertex $s$ in $D$. Let $V_s$ be the set of vertices $v$ such that there exist a dipath from $s$ to $v$ in $D$. First, we do a topological ordering of the vertices in $V_s$ considering the subgraph induced by its arcs. Therefore, we can put the vertices of $V_s$ in an horizontal line in such a way that the arcs are all directed from the left to the right (it is always possible to do such topological ordering in kinship networks because the graph induced by its arcs is acyclic). Create a set of the partition $A_s$. Add the vertex $s$ in $A_s$ and make $s$ be its representative. Next, repeat the following steps for each vertex $v$ ($\neq s$) on the topological ordering:

**a.** if $v$ is child of $s$ or if $v$ has parents in different sets of the partition, then create a new set of the partition $A_v$, add $v$ in $A_v$, and make $v$ be its representative;

**b.** otherwise, that is, if $v$ is not child of $s$ and all parents of $v$ are in a single set of the partition, say $A_z$, then add $v$ in $A_z$.

We can use induction over the topological order to argue that the algorithm above is correct. Consider what the algorithm do in part **a**. If $v$ is child of $s$, then $v$ should stay in $A_v$ ($\neq A_s$) since we have that $s$ is a junction of $s$ and $v$ ($s$ is in $J_{sv}$). The result is obtained when we create $A_v$ and make $v$ be its representative. If $v$ is child of vertices $p_1$ and $p_2$ and they are in diferent sets, say $p_1$ in $A_u$ and $p_2$ in $A_w$, then by induction $s$ is in $J_{p_1 p_2}$. So, there are internally vertex-disjoint dipaths $P$ from $s$ to $p_1$ and $Q$ from $s$ to $p_2$. By the topological order, $v$ is not in $P$ neither $Q$. Let $x$ be a vertex in $A_u$. Then, $s$ is in $J_{xv}$ since $P$ and $Q$ concatenated with arc $p_2 \to v$ are vertices-disjoint dipaths. In a similar way, we can show that $s$ is in $J_{xv}$ when $x$ is in $A_w$. Now, let $x$ is in $A_y$ ($\neq A_u$ and $\neq A_w$). By induction, there are vertex-disjoint dipaths $P$ from $s$ to $x$ and $Q$ from $s$ to $p_2$ (for example). Thus, the dipaths $P$ and $Q$ concatenated with the arc $p_2 \to v$ are vertex-disjoint dipaths, and so $s$ is in $J_{xv}$. Therefore, we can create a new set $A_v$ and add $v$ to $A_v$ as its representative to obtain the result.

Now consider what the algorithm do in part **b**. Suppose that the parents of $v$, say $p_1, p_2, \ldots, p_k$, are all in a set $A_z$. Let $x$ in $A_z$ ($x$ can be one of the parents of $v$). By induction, $s$ is not in $J_{xp_i}$ ($i = 1, 2, \ldots, k$). Thereby, for any pair $x$ and $p_i$ we have that any pair of dipaths from $s$ to $x$ and from $s$ to $p_i$ contains at least a commum vertex. This show that $s$ is not in $J_{xv}$. Now, let $x$ in $A_u$ ($\neq A_z$). By induction, $s$ is in $J_{xp_i}$ ($i = 1, 2, \ldots, k$). Therefore, there are vertex-disjoint dipaths $P$ from $s$ to $x$ and $Q$ from $s$ to $p_i$. Since $v$ cannot occur in $P$ and in $Q$ (by the topological order) we have that $P$ and $Q$ concatenated with the arc $p_i \to v$ are vertex-disjoint dipaths. Thus, $s$ is in $J_{vx}$. Therefore, add $v$ in $A_z$ produces the result.

As we said before, we can find in [13] and [14] different ways to proof that the partition built by the cited algorithm has the desired property. However, the way presented here is the simplest one. It is also important to note that such partition is build in $O(n + m)$ time where $n$ is the size of the set of vertices and $m$ is the size of the set of arcs of the input kinship network.

To finish step 1, considering each pair of vertices $u$ and $v$, we add $s$ in $J_{uv}$ whether $u$ and $v$ are in different sets of the partition. We repeat all the previous procedure for each vertex $x$ in $D$, building in this way, the sets of all junctions of all pairs of vertices in $F$.

### *3.2. Step 2*

In this step, we will talk about the number of possibles sorted sets with $k$ marriages. If $t$ is the total number of marriages in the network, then the number is $t \times (t-1) \times (t-2) \times \ldots \times (t-k+1)$ since we have $t$ possibilities for the first marriage in the sorted set, $t-1$ possibilities for the second marriage, (and each possible second marriage choice can be combined with each possible first marriage choice), and so on, up to the $k$-th marriage. In a similar way, we can choose $k$ marriages from $t$, and then consider all rearrangement of the $k$ marriages chosen. Next step is applied for each $k$ sorted marriage set.

*3.3. Step 3*

In this step, we assume a $k$ sorted marriage set $C = \{(u_1, v_1), \ldots, (u_k, v_k)\}$. Moreover, we know the sets of all junctions of all pairs in $F$ (given by step 1). Let us suppose that $k > 1$. Therefore, the pairs of vertices (in $C$) $u_i$ and $u_j$, $u_i$ and $v_j$, and $v_i$ and $v_j$ ($i \neq j$) are in $F$. Fixed the ordering of $k$ marriages given by the set $C = \{(u_1, v_1), \ldots, (u_k, v_k)\}$, note that finding all rings which contain such marriages (in that order) imply to verify the presence of junctions between vertices of consecutive marriages in $C$. For example, to find all A3C3 which contain the marriages $(u_1, v_1)$, $(u_2, v_2)$ and $(u_3, v_3)$, in this order, we have to verify the junctions of 8 possibles pairs of vertices. Fig. 5 illustrates all these cases.

Still considering the previous example (to find all A3C3 rings), and now fixed a possible way to combine pairs of vertices in 3 marriages in the sorted set $C = \{(u_1, v_1)$, $(u_2, v_2)$ and $(u_3, v_3)\}$, the procedure in step 3 realizes the following (considering that the way chosen to combine pairs was $\{v_1, v_2\}$, $\{u_2, u_3\}$ and $\{u_1, v_3\}$, the same way of the second ring in Fig. 5):

- **a.** for each junction $s$ of $v_1$ and $v_2$, find all internally vertex-disjoint dipaths from $s$ to $v_1$ and from $s$ to $v_2$. Denote such dipaths by $P^1_{sv_1}, P^2_{sv_1} \ldots, P^{n_{v_1}}_{sv_1}$ and $Q^1_{sv_2}, Q^2_{sv_2} \ldots, Q^{n_{v_2}}_{sv_2}$.
- **b.** for each junction $s$ of $u_2$ and $u_3$, find all internally vertex-disjoint dipaths from $s$ to $u_2$ and from $s$ to $u_3$. Denote such dipaths by $P^1_{su_2}, P^2_{su_2} \ldots, P^{n_{u_2}}_{su_2}$ and $Q^1_{su_3}, Q^2_{su_3} \ldots, Q^{n_{u_3}}_{su_3}$.
- **c.** for each junction $s$ of $u_1$ and $v_3$, find all internally vertex-disjoint dipaths from $s$ to $u_1$ and from $s$ to $v_3$. Denote such dipaths by $P^1_{su_1}, P^2_{su_1} \ldots, P^{n_{u_1}}_{su_1}$ and $Q^1_{sv_3}, Q^2_{sv_3} \ldots, Q^{n_{v_3}}_{sv_3}$.
- **d.** for each possible choice of 3 dipaths $P^i_{sv_1}, P^{i'}_{su_2}, P^{i''}_{su_1}$ and 3 dipaths $Q^j_{sv_2}, Q^{j'}_{su_3}, Q^{j''}_{sv_3}$, if the intersection among the chosen dipaths is empty, then they form an A3C3 ring. Therefore, store it.

In general, considering a sorted set of $k$ marriages, the number of combinations of pairs of this set is $T(k) = 1$, case $k = 1$; $T(k) = 2$, case $k = 2$; $T(k) = 2(4^{\frac{k-1}{2}})$, case $k \geq 3$ and odd; or $T(k) = 4^{\frac{k}{2}}$, case $k \geq 4$ and even. This happens because for each 2 consecutive marriages, there are 4 ways to combine the vertices in these marriages. If $k$ is odd ($\geq 3$), then the last marriage closes a ring with 2 more different ways. Therefore, $T(k) = 2^k$ for $k \geq 3$.

Let the depth of a kinship network be the number of vertices of the biggest dipath. If the depth of a kinship network is small, then the time spend to find the dipaths in itens **a**, **b**, **c** described above and the time

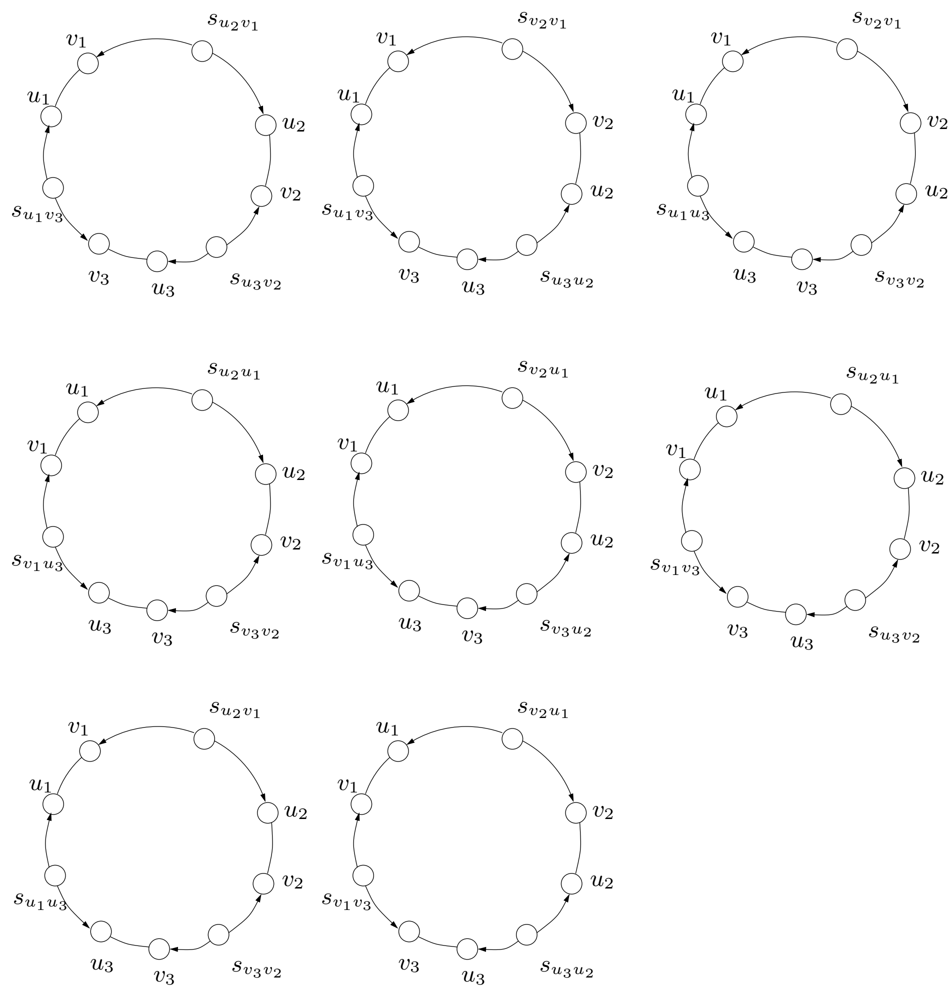


Figure 5: The 8 ways to combine pairs of vertices in 3 marriages in the order $(u_1, v_1)$, $(u_2, v_2)$ and $(u_3, v_3)$.

spend in item **d** are tolerated. Next section, we describe our experience applying steps 1, 2 and 3 in eight real kinship networks. We also describe some practical aspects of our research and some preliminary results.

## 4. Practical aspects and experimental results

We start this section talking a bit more about *exchange rules* when the exchange is represented by a *marriage* between individuals. The marriages of some societies are conducted by a *positive marriage rules*. An example is the following positive rule: “a man always marry to a mother's brother's daughter”. This provides a certain structure to the corresponding kinship network. In the *Alliance Theory*, the positive marriage rules create *elementary structures* and its representation on a kinship network is given by a subnetwork. The composition of elementary structures provides us a structured kinship network and, consequently, a structured social relationship among individuals.

Other societies make marriages impossible among individuals by *negative marriage rules*. For example, a negative rule could be “a man **must not** marry to a mother's brother's daughter”, the opposite of the previous positive marriage rule. In this case, a man has many possibilities to marry, and it leaves the corresponding kinship network less structured. However, marriages still exist among individuals. Is there any hidden rule? Is it possible to find any “structure” in societies where the marriages rules are negatives? If so, once again, we will have now a (semi-)structured kinship network, and therefore a (semi-)structured social life people. The answer of the previous questions are given by a kin data analysis. We are going to focus on the *ring analysis* in kinship networks.

In any network, the simplest structure which represents the individual exchange among different families is the ring. Note that any elementary structure in a network has to contain a ring. Therefore, an important task is to develop fast methods to find rings in networks. Moreover, a fixed marriage could appear in different rings. For example, two married individuals $u$ and $v$ could be linked by dipaths from two different commom ancestors. The set of all rings that have fixed marriages on them is called as *implex* [7]. Given a fixed marriage set $M$, we denote by A$p$C$q$ $M$-implex the set of A$p$C$q$ rings containing the marriages $M$ and the size of $M$ is $p$ ($|M| = p$). We have $p = q$ when there is no commom individual as an individual in some marriage in $M$. We have seen in practice that $p$ and $q$ is less than or equal to 3. See Figure 6 for an example of A2C1 $M$-implex where $M = \{u - v, w - v\}$.

An implex is an attempt to understant the exchange among different families in societies where the marriages are conducted by negative rules. However, in this work, we concentrate the computational experiments in statistical measures of some kinship networks.

During our experiments, we worked with 8 empirical kinship networks: Arara, Xavante, Irantxe-Myky, Zoró, Enawenê-Nawê, Deni, Arapium and Krahô. Table 1 informs for each those kinship network the number of vertices (or individuals) $n$, the number of arcs (or the consanguinity connections) $m$, the number of edges (or affinity connections) $t$, and the size of a longest

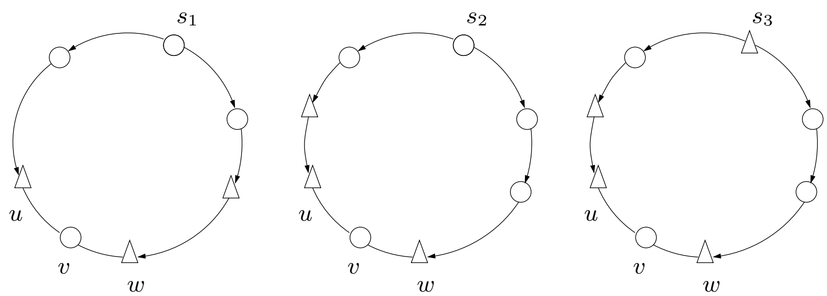


Figure 6: An A2C1 $M$-implex where $M = \{u - v, w - v\}$.

dipath $l$. We did not found publications related to the Zoró network. We refer it by initials unp (unpublished).

Table 1: Empirical kinship networks

| **Networks** | **References** | $n$ | $m$ | $t$ | $l$ |
|---|---|---|---|---|---|
| Arara | [15] | 105 | 197 | 48 | 5 |
| Xavante | [16] | 459 | 713 | 254 | 6 |
| Irantxe-Myky | [17] | 618 | 1003 | 177 | 6 |
| Zoró | unp | 755 | 1224 | 201 | 6 |
| Enawenê-Nawê | [18] | 789 | 1368 | 170 | 6 |
| Arapium | [19] | 1214 | 1792 | 291 | 6 |
| Deni | [20] | 875 | 1589 | 333 | 7 |
| Krahô | [21] | 1031 | 1793 | 379 | 8 |

We will concentrate in the efficiency of the methods that enumerate all rings from the empirical kinship networks. Unfortunately, we can show that rings enumeration can take an exponential time with respect to the network size, since there could be an exponential number of rings to enumerate. Despite that, our method was able to enumerate the rings and implexes from the networks cited[5].

Figure 7 draws a boxplot[6] for each empirical network as data representing the time spent to output an A1C1 ring during a total tool running time.

[5]For Deni and Krahô networks, the enumeration of A3C3 rings was stoped after reaching 3.000.000 rings (see Table 3).

[6]All boxplot whiskers are 1.5×IQR (Interquartile range).

The time spent to decide that a marriage do not be in any A1C1 ring was contabilized. Beyond that, we can observe good results which can be explain by the small size of a longest dipath for all networks analysed (see last column in Table 1). Therefore, restricted to network with small longest dipaths, the task to output A1C1 rings could be done quickly (Step 3 from subsection 3.3 applied to A1C1 rings).

In Figure 8 each data represents the time spent to list an A1C1 implex. Remember that an implex is the set of all rings for a fixed marriage set $M$ [7]. In this case, given marriage set $M$, we measuared only the time spent to list a *non-empty* $M$-implex. We decide did not present the time for *empty* $M$-implex, however, we point out Figure 7 once again to confirm that this time is small.

Tables 2 and 3 shows, for each empirical network, the number of rings and different implexes founded. We have to remark that 79% of marriages from Arara network are consanguinity marriages (A1C1 case). These percentage drop to 21% for Irantxe-Myky marriages, 48% for Deni marriages and 27% for Krahô marriages. For the other networks, less than 15% of marriages are consanguinity marriages. Two marriages in Arara network have 60% of chance to be a consanguinity relinking marriage (A2C2 case). The chance drop to 18% to Deni network and to 9.5% to Krahô Deni. For the other networks, these chance drop to less than 6%. Three marriages in Arara network have 30% chance to be an affinal relinking marriage (A3C3 case).

Table 2: Number of rings ($R$), implexes ($I$) and rings by implex (A1C1 and A2C2 cases)

| | A1C1 | | | A2C2 | | |
|---|---|---|---|---|---|---|
| Networks | $R$ | $I$ | $R/I$ | $R$ | $I$ | $R/I$ |
| Arara | 163 | 38 | 4.29 | 14.645 | 1.357 | 10.79 |
| Xavante | 56 | 26 | 2.15 | 7.455 | 1.482 | 5.03 |
| Irantxe-Myky | 120 | 38 | 3.16 | 6.390 | 1.541 | 4.15 |
| Zoró | 57 | 26 | 2.19 | 2.976 | 863 | 3.45 |
| Enawenê-Nawê | 16 | 8 | 2 | 2.539 | 606 | 4.19 |
| Arapium | 78 | 29 | 2.69 | 1.902 | 464 | 4.10 |
| Deni | 918 | 159 | 5.77 | 320.155 | 20.199 | 15.85 |
| Krahô | 812 | 102 | 7.96 | 353.542 | 13.628 | 25.94 |

Figures 9 and 10 show, respectively, the time spent to list A2C2 and A3C3 implexes. As in the case of A1C1 implexes (Figure 8), the data in Figures 9 and 10 represent the time spent to list a non-empty A2C2 and A3C3 $M$-implex for all corresponding set $M$. In these figures, note an

Table 3: Number of rings ($R$), implexes ($I$) and rings by implex (A3C3 case)

| | A3C3 | | |
|---|---|---|---|
| Networks | $R$ | $I$ | $R/I$ |
| Arara | 936.176 | 50.014 | 18.72 |
| Xavante | 599.585 | 68.394 | 8.77 |
| Irantxe-Myky | 293.374 | 44.268 | 6.63 |
| Zoró | 146.758 | 25.755 | 5.70 |
| Enawenê-Nawê | 144.387 | 19.112 | 7.55 |
| Arapium | 31.712 | 5.768 | 5.50 |
| Deni | $\geq$ 3.000.000 | 27.586 | 108.75 |
| Krahô | $\geq$ 3.000.000 | 104.804 | 28.62 |

interesting inversion in the boxplot for Deni e Krahô networks. Almost 75% of the A2C2 implexes from Deni are below the median A2C2 implex from Krahô (See Figure 9). However, 75% of the A3C3 implexes from Krahô are below the median A3C3 implex from Deni (see Figure 10). We explain these behavior as the following: the mean of A2C2 rings by implex are 15.85 (for Deni) and 25.94 (for Krahô), whereas the mean of A3C3 rings by implex are 108.75 (for Deni) and 28.62 (for Krahô). So, listing an A2C2 implex (resp. A3C3 implex) for Deni (resp. Krahô) could be done faster than listing an A2C2 implex (resp. A3C3 implex) for Krahô (resp. Deni).

To finish this section, we would like to describe the following observation. Consider monogamic networks where each individual or does not have parent; or has two parents. Let us name them as 0/2 *monogamic networks*. In these networks, an A$k$C$k$ ring has (fakes rings??)

## 5. Finding rings, finding one ring and its complexity

Previously, we described a way to enumerate all A$k$C$k$ rings. Now we will show that we can find all $AkCl$ ($k \neq l$) through of any algorithm that find all $AkCk$. Let $R$ be an $AkCl$ ring and $k \geq l$, and suppose that $C$ is the sorted marriage set in $R$. So, we can see in Fig. 14 that the number of common vertices on the edges in $C$ is equal to $k - l$. For example, the number of common vertices on the edges in an $A3C2$ is 1, whereas in an $A3C3$ is 0 (see Fig. 14). Next, we will show that an algorithm $A$ which finds all $AkCk$ ring over an sorted marriage set (note that the number of edges and the number of junctions are equal) can be used to find all $AkCl$ ring ($k \neq l$) over a new sorted marriage set $C'$.

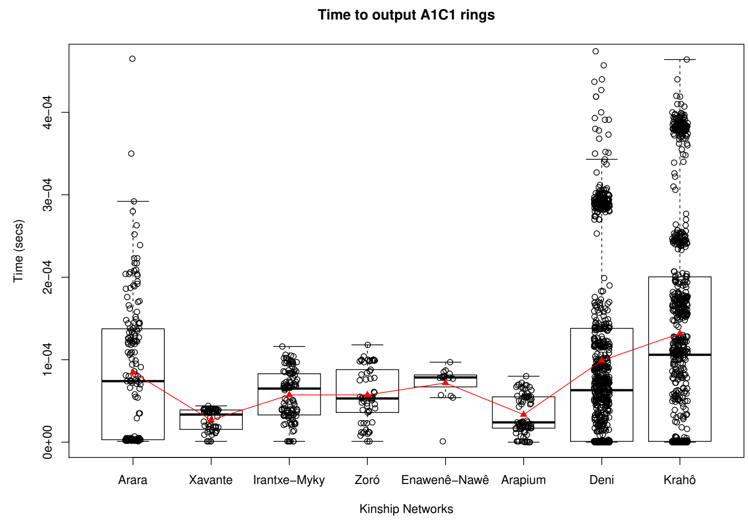


Figure 7: Time to output an A1C1 rings. The line indicates the mean.

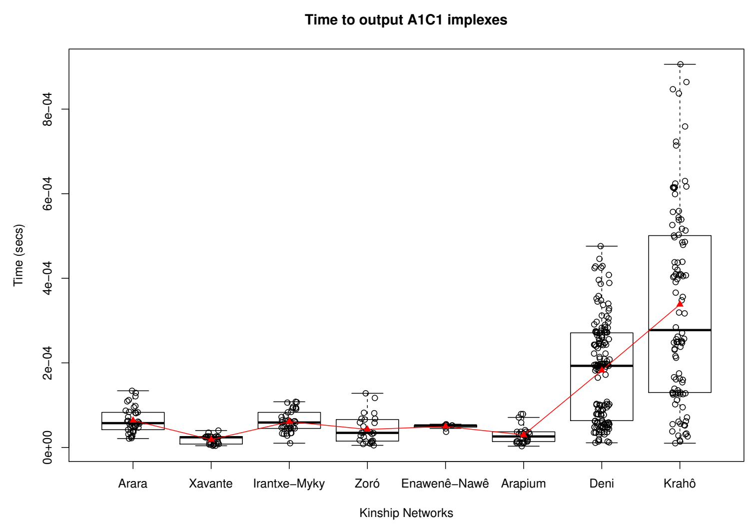


Figure 8: Time to list A1C1 implexes for Xavante, Irantxe-Myky, Zoró, Enawenê-Nawê and Arapium. The triangles indicate the mean.

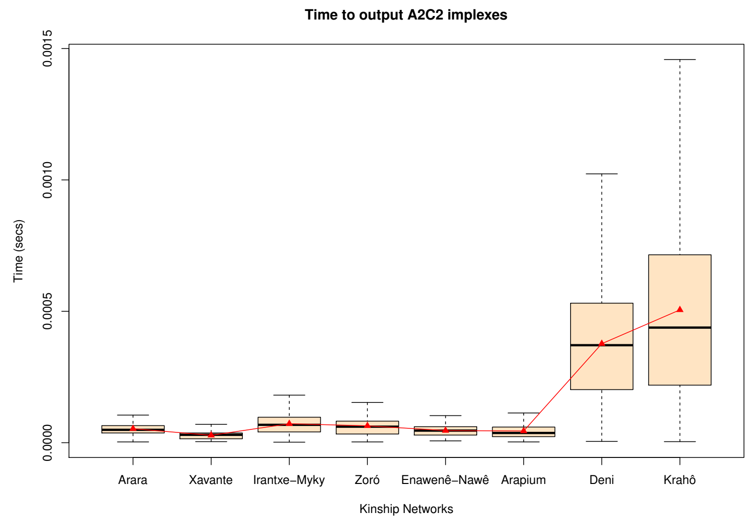


Figure 9: Time to list A2C2 implexes for Xavante, Irantxe-Myky, Zoró, Enawenê-Nawê and Arapium. The triangles indicate the mean.

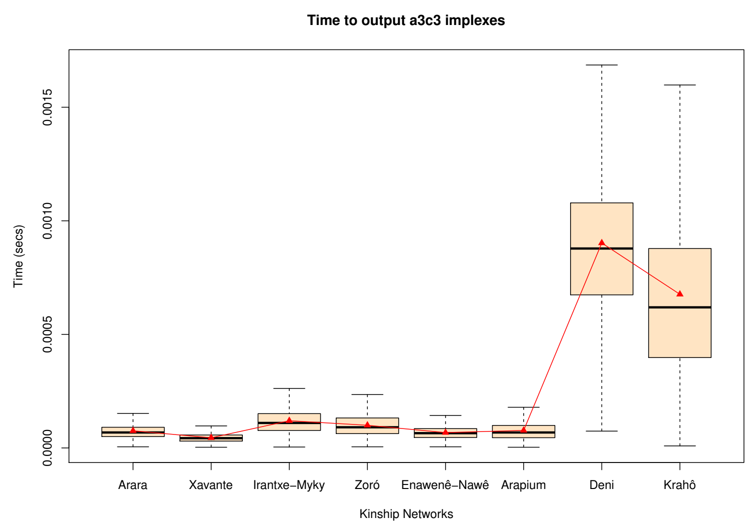


Figure 10: Time to list A3C3 implexes for Xavante, Irantxe-Myky, Zoró, Enawenê-Nawê and Arapium. The triangles indicate the average time.

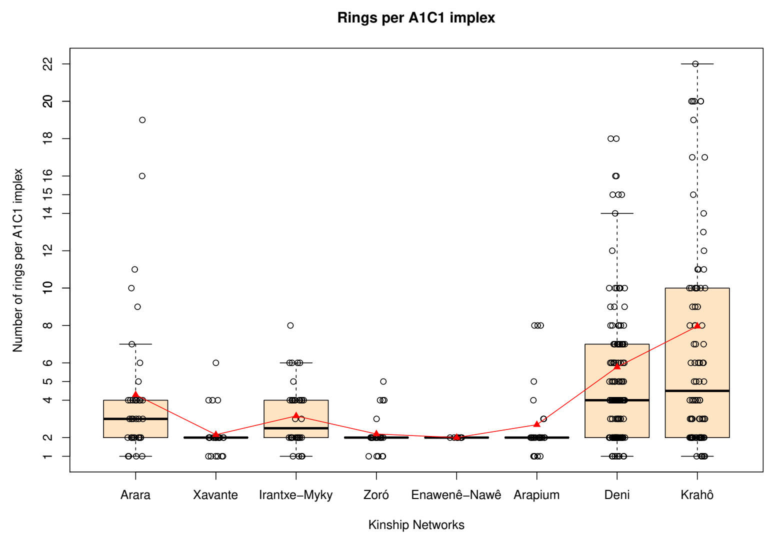


Figure 11: Rings per A1C1 implex. The triangles indicate the mean.

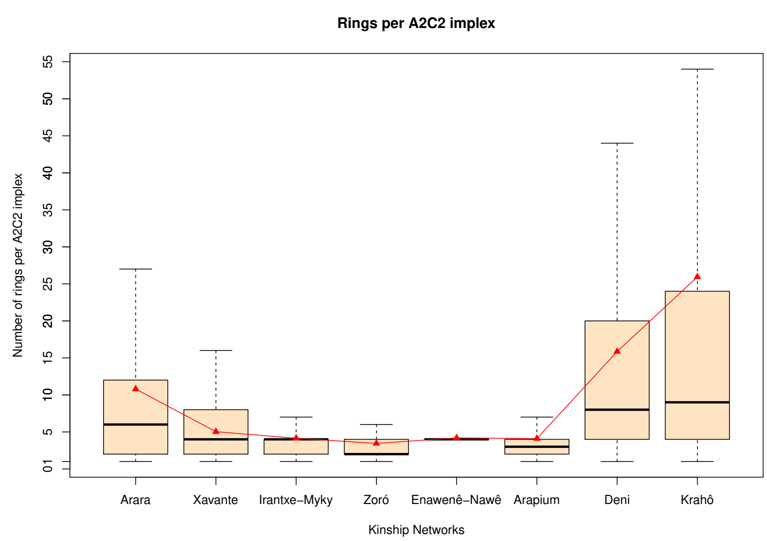


Figure 12: Rings per A2C2 implex. The triangles indicate the mean.

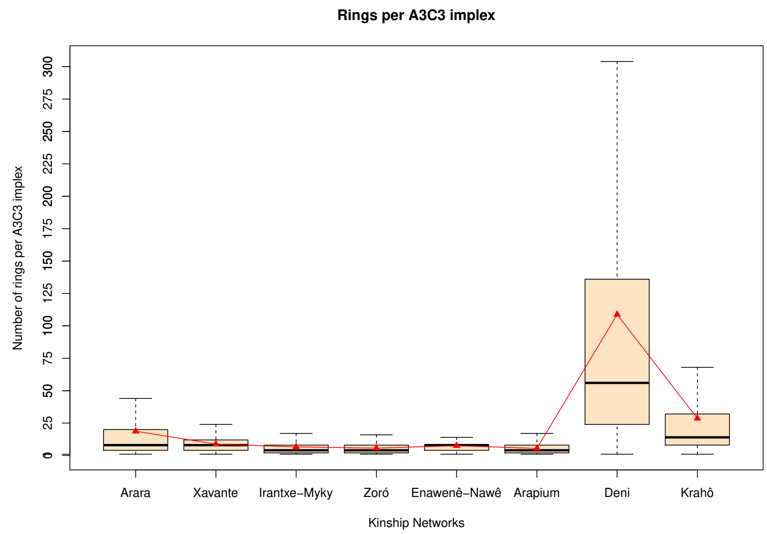


Figure 13: Rings per A3C3 implex. The triangles indicate the mean.

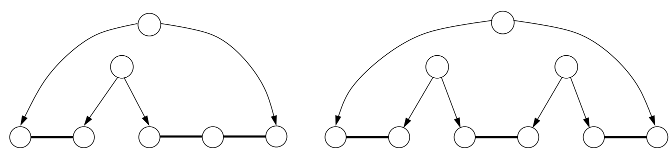

Figure 14: An $A3C2$ and $A3C3$.

We start defining a *children triad* which is a digraph formed by three vertices $s$, $u$ and $v$, and two arcs $s \to u$ and $s \to v$. We denote such digraph by $CT(s, u, v)$. Given a kinship network $H$ and a sorted marriage set $C$ with $k$ marriages and $p$ shared vertices by edges in $C$ ($p \geq 1$), we construct a new kinship network $H'$, a new sorted set $C'$, and apply an algorithm which finds all A$k$C$k$ rings on $H'$ and $C'$ such that:

> *any $AkCk$ ring over $C'$ in $H'$, corresponds to an $AkCl$ ring over $C$ in the original kinship network $H$.*

The construction can be followed in Fig. 15. First, we denote the common vertices in $C$ by $u_1 = v_1, u_2 = v_2, \ldots, u_p = v_p$. Now, we create a children triad $CT(s'_i, u'_i, v'_i)$ for each vertex $u_i = v_i$ $(i = 1, \ldots p)$. Finally, we replace each vertex $u_i = v_i$ by its corresponding children triad $CT(s'_i, u'_i, v'_i)$

as described in the following steps:

1. initialize $H^1 \leftarrow H$ and $C^1 \leftarrow C$,
2. for each $i = 1, \ldots, p$,
   (a) replace the edges $u_i - x$ and $y - v_i$ in $C^i$ by $u'_i - x$ and $y - v'_i$, respectively, and
   (b) update $H^{i+1} \leftarrow H^i$ and $C^{i+1} \leftarrow C^i$.
3. update $H' \leftarrow H^i$ and $C' \leftarrow C^i$.

It is simple to prove a correspondence of $AkCl$ rings in $H$ and $AkCk$ rings in $H'$. Therefore, we can apply any algorithm that finds all $AkCk$ ring over $H'$ and $C'$ to obtain all $AkCl$ rings over $H$ and $C$.

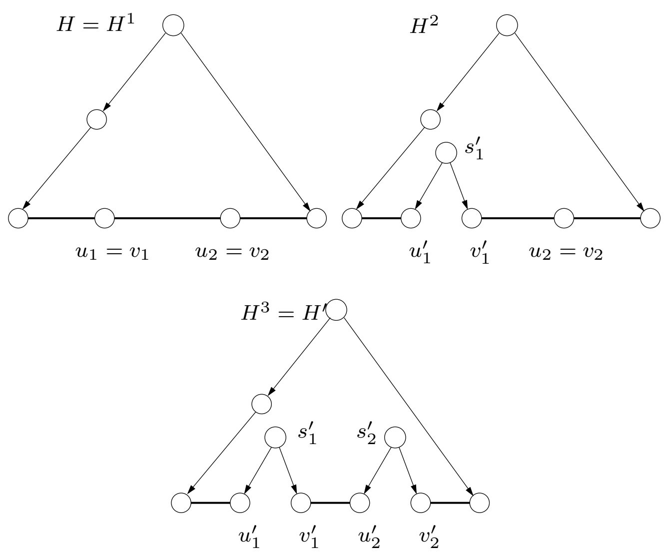


Figure 15: Example of the correspondence between an $A3C1$ over $H$ and $C$ and $A3C3$ over $H'$ and $C'$.

Now, we will proof that the problem of finding only one A$k$C$k$ over a given kinship network $H$ and a sorted marriage set $C$ is NP-hard. The reduction is from the $k$ *vertex-disjoint dipath* problem stated below (it is NP-hard due to Even, Itai and Shamir [22]).

> The $k$ vertex-disjoint dipath problem: Given an acyclic digraph $D$, and $k$ ordered pairs of vertices $(s_1, t_1), \ldots, (s_k, t_k)$, decide whether there are vertex-disjoint dipaths $P_i$ from $s_i$ to $t_i$ for $i = 1, \ldots, k$.

Given an acyclic digraph $D$, and $k$ ordered pairs of vertices $(s_1, t_1), \dots, (s_k, t_k)$, we construct a new acyclic digraph $D'$ which initially is a copy of $D$. After, we add in $D'$ a new vertex $t'_i$, and a new arc $s_i \to t'_i$ for $i = 1, \dots, k$. We add in $D'$ a sorted set of edges $C = (w_1, \dots, w_k)$, where $w_1 = t_1 - t'_k$ and $w_i = t_i - t'_{i-1}$ for $i = 2, \dots, k$. These edges added in $D'$ makes it a mixed graph. Now we show that there are vertex-disjoint dipaths $P_i$ from $s_i$ to $t_i$ in $D$ for $i = 1, \dots, k$ if and only if there is a $AkCk$ ring over $D'$ and $C$.

If $P_i$ is a vertex-disjoint dipath in $D$ from $s_i$ to $t_i$ for $i = 1, \dots, k$, then we can make $\mathcal{P} = \{P_1, Q_1, \dots, P_k, Q_k\}$ where $Q_i$ is the arc $s_i \to t'_i$ for $i = 1, \dots, k$ to obtain an $AkCk$ ring over $D'$ and $C$ (see Fig. 16).

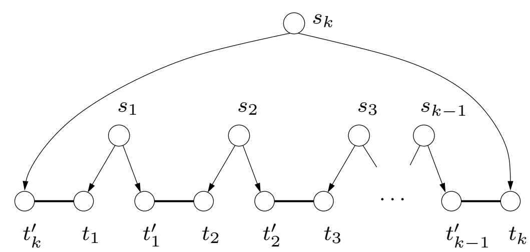


Figure 16: An $AkCk$ ring over $D'$ and $C$.

Now, consider an $AkCk$ ring $R$ over $D'$ and $C$ where $\mathcal{P} = \{P_1, Q_1, \dots, P_k, Q_k\}$ is the set of dipaths in $R$. By definition of ring, $P_i$ and $Q_i$ start in a junction of vertices $t_i$ and $t'_i$ for $i = 1, \dots, k$. By the construction of $D'$, the unique junction of vertices $t_i$ and $t'_i$ is $s_i$. Thus, the dipaths $P_1, \dots, P_k$ are vertex-disjoint which link the pairs $(s_1, t_1), \dots, (s_k, t_k)$ in $D$. $\square$

## 6. Chromatic rings on vertices

The individuals of some societies belong to determined groups. For example, each individual from *Enawenê-Nawê* people belongs to a clan (see `http://pib.socioambiental.org/en/povo/enawene-nawe/485`). This *partitioning* of the individual set can be represented by a color for each clan and, therefore, a color for each individual. In these cases, *chromatic rings* start to occur. We say that a ring is *p-chromatic* when it has $p$ colors, and each consanguinity line, from junctions to individuals married, has a determined color. For example, in Fig. 17 (a), the A2C1 ring is 1-chromatic, since it has 1 color and the consanguinity lines from junction $s_1$ to vertices $u$ and $w$, and the vertex $v$, they have the same color; in Fig. 17 (b), the A3C3 ring is 3-chromatic, since it has 3 colors and the consanguinity lines from $s_1$ to $u$ and $w$, from $s_2$ to $z$ and $y$, and from $s_3$ to $x$ and $v$ they have the same color (red, blue and green, respectively); lastly, in Fig. 17 (c), the A3C3 ring is 2-chromatic.

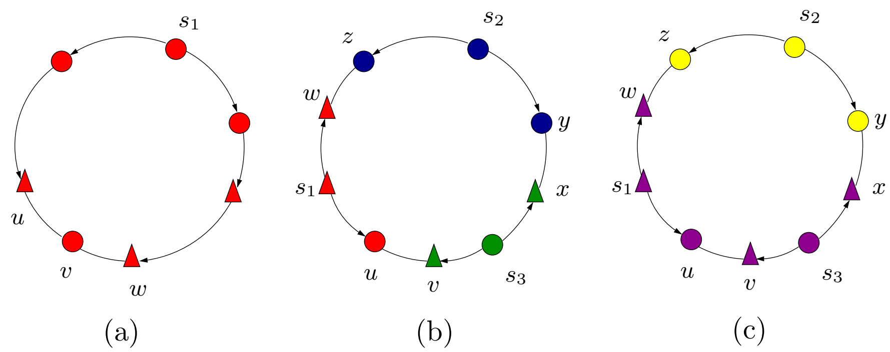


Figure 17: Chromatic rings.

The motivation to find chromatic rings in kinship networks comes by the interest to analyze matrimonial rules from the individual groups point of view. For example, in Fig. 17 (b), the individual $v$ from green group, when he/she marry to individual $u$ from red group, he/she *relinking* the alliances among groups green, blue and red. In Fig. 17 (a), the affinity connections are restricted to a determined group.

In a formal way, we state some problems.

**P.1:** Given a kinship network $D$ with all vertices colored and a sorted marriage set $C$, find only one chromatic ring $R$ over $D$ and $C$.

**P.2:** Given a kinship network $D$ with all vertices colored, a sorted marriage set $C$ and a list of colors $L$, find only one chromatic ring $R$ over $D$ and $C$ where $R$ has only colors in $L$.

**P.3:** Given a kinship network $D$ with all vertices colored, a sorted marriage set $C$ and an positive integer number $p$, find only one $p$-chromatic ring $R$ over $D$ and $C$.

Up to now, we do not know any tool that finds chromatic rings in kinship networks. The construction of tools able to find such rings is one of the challenges of this area since all these problems are NP-hard. The reduction is from the problem previously shown to be hard. To see this result, we just color all vertices of a given network with a unique color. So, a solution to problems **P.1**, **P.2** and **P.3** is also a solution to the previously one.

## 7. Chromatic rings on arcs and edges

Usually, a kinship network is formed by connections of affinity and consanguinity. However, some people relates individuals by other ways. One example of this is the connections of *nomination* and *formal friendship* from Krahô people (`http://pib.socioambiental.org/en/povo/kraho/443`). When an individual $u$ *nominates* another individual $v$, then we have a *connection of nomination* between individuals $u$ and $v$. The nomination of an individual $u$ implies in the *connection of formal friendship* between $u$ and other individuals with particular names. Summarizing, new connections are added to the kinship network obeying the following rules:

- the connections of nomination are directed, that is, there exists an arc in the network from $u$ to $v$, whether the individual $u$ nominates the individual $v$;
- the connections of formal friendship are represented by edges. A new edge is created in the network between $u$ and $v$, whether individuals $u$ and $v$ are formal friends to each other.

Moreover, as well as the letters F, M, S, D, H, and W are associated to connections of consanguinity and affinity[7], the letters K, T, I, U, O and P are associated to connection of nomination and formal friendship (proposed in [23]).

- Connections of nomination:
  - K (*Keti* in Krahô language) to male nominator;
  - T (*Tii*) to female nominator;
  - I (*Itanpú*) to male nominated; and
  - U (*Itanpú*) to female nominated;
- Connections of formal friendship:
  - O (*Hõpin*) to male formal friend; and
  - P (*Pintxwoi*) to female formal friend.

[7] Consanguinity: F (*Father*), M (*Mother*), S (*Son*) and D (*Daughter*); Affinity: H (*Husband*) and W (*Wife*).

The connections of nomination do not cause an oriented cycle in the network since an individual of a determined generation nominates individuals of posterior generation (and never previous generation). Like before, a ring in these networks is a cycle without parental triad. The motivation to find rings in these networks comes from the interest to analyze the influence of the news connections on the social behavior of this people.

It is important to observe that kinship networks only composed by connections of consanguinity and affinity, there is no rings with two arcs entering a vertex since, in that case, the absence of parental triad implies in the absence of vertices of in-degree equal to 2. However, in the networks with these news connections, vertices with in-degree equal to 2 can occur in rings. See the ring in Fig. 18 that illustrate a case.

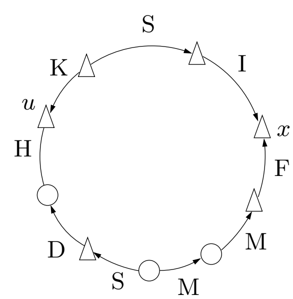


Figure 18: A ring KSIFMMSDH (reading clockwise starting and finishing in vertex $u$). Note that the ring has a vertex with in-degree equal to 2 (the vertex $x$ with consecutive connections I and F).

To represent the different connections in Krahô network, we can assign colors to its arcs and edges. For example, the consanguinity connections can be blue, the nomination connections can be yellow, the affinity connections can be red and the formal friendship connections can be green color. Somehow, the rings in these cases are also chromatic on the arcs and edges. Formally, we can state the following problem.

**P.4:** Given a kinship network $D$ with colors on the arcs and edges, a sorted marriage set $C$ and a color $c$, find a ring over $D$ and $C$ whose number of arcs with color $c$ is at least 1.

Once again, we can reduce the problem of finding one ring in kinship networks to **P.4** problem. So, it also is a NP-hard problem. What we have to do is only colored all arcs and edges with a single color.

We believe that the development of algorithms to enumerate chromatic rings in networks with colors in arcs and edges is an important contribution to the Structural Anthropology area. The retrieval of such rings implies in finding rings with vertices with in-degree equal to 2. It does not occur when the arcs and edges represent only consanguinity and affinity. Moreover, a solution to the rings with new connections, as the new connections from the Krahô people, can provide to the network analyst new ways to evaluate, to verify and to justify the social behavior and social organization of a people.

## 8. Conclusion

In this text we presented a way to enumerate A$k$C$k$ rings in kinship networks. The preprocessing of the network and the decomposition of the task in three steps were fundamental to the successful of the enumeration. The steps are: 1. Find the sets of all junctions of all pairs of vertices; 2. Build a sorted marriage set with $k$ fixed marriages $C = \{(u_1, v_1), \ldots, (u_k, v_k)\}$; and 3. Use a procedure that enumerates A$k$C$k$ rings involving the $k$ fixed marriages and in the order which they appear in the set. To enumerate all the rings of a network, we have to repeat step 3 for all possible sorted marriage set with $k$ marriages.

We also present some problems that we believe to be interesting as to the Anthropology as to the Computing: finding chromatic rings with colors on vertices, edges and arcs. We showed that all these problems are NP-hard.

## References


[1] O. Ore, Sex in graphs, Proceedings of the American Mathematical Society (1960) 533–539.

[2] D. R. White, P. Jorion, Kinship networks and discrete structure theory: Applications and implications, Social Networks 18 (3) (1996) 267–314.

[3] K. Hamberger, M. Houseman, D. R. White, Kinship network analysis, The sage handbook of social network analysis (2011) 533–549.

[4] C. Lévi-Strauss, Elementary Structures of Kinship, Vol. 340, Beacon Press, 1969.

[5] L. Dumont, Introducción a dos Teorías de la Antropología Social, Editorial Anagrama, 1975.

[6] D. R. White, Ring cohesion theory in marriage and social networks, Mathématiques et sciences humaines. Mathematics and social sciences 168.

[7] M. F. d. Silva, J. dal Poz, Maqpar: a homemade tool for the study of kinship networks, Vibrant 6 (2) (2009) 29–51.

[8] D. R. White, V. Batagelj, A. Mrvar, Anthropology: Analyzing large kinship and marriage networks with pgraph and pajek, Social Science Computer Review 17 (3) (1999) 245–274.

[9] K. Hamberger, M. Houseman, I. Daillant, D. R. White, L. Barry, Matrimonial ring structures, Mathematics and Social Science 168 (2004) 83–119.

[10] V. Batagelj, A. Mrvar, Analysis of kinship relations with pajek, Social Science Computer Review.

[11] K. Hamberger, Matrimonial circuits in kinship networks: Calculation, enumeration and census, Social Networks 33 (2) (2011) 113–128.

[12] C. E. Ferreira, A. J. P. Franco, M. F. d. Silva, Finding matrimonial circuits in some amerindian kinship networks: An experimental study, in: 10th International Conference on e-Science, Vol. 1, IEEE, 2014, pp. 73–80.

[13] C. E. Ferreira, A. J. P. Franco, Facets of Combinatorial Optimization - Festschrift for Martin Grötschel, Springer Verlag, 2013, Ch. Algorithms for junctions in acyclic digraphs.

[14] C. E. Ferreira, A. J. P. Franco, A min–max relation in flowgraphs and some applications, Discrete Applied Mathematics(To appear).

[15] M. Teixeira-Pinto, Ieipari: sacrifício e vida social entre os índios Arara (Caribe), Editora Hucitec, 1997.

[16] D. Maybury Lewis, Akwé-Shavante Society, Oxford University Press, 1967.

[17] A. C. V. Bueno, Fios de memórias. Um estudo sobre parentesco e história a partir da construção da genealogia manoki (irantxe), Ph.D. thesis, Univ. of São Paulo (2015).

[18] M. F. Silva, Liga dos Enawenê-Nawê: um estudo da aliança de casamento na Amazônia Meridional (2012).

[19] L. M. Lima, No Arapiuns, entre verdadeiros e - ranas: sobre os espaços, as lógicas, as organizações e os movimentos do político, Ph.D. thesis, Univ. of São Paulo (2015).

[20] M. P. Florido, Os Deni do Cuniuá: um estudo do parentesco, Ph.D. thesis, Univ. of São Paulo (2013).

[21] J. C. Melatti, O Sistema Social Craô, Ph.D. thesis, Univ. of São Paulo (1970).

[22] S. Even, A. Itai, A. Shamir, On the complexity of time table and multicommodity flow problems, in: Proc. 16th Annual Symp. on Found. of Computer Science, SFCS '75, Washington, DC, USA, 1975, pp. 184–193. doi:10.1109/SFCS.1975.21.

[23] C. M. d. O. Paulino, A rede Mehĩ: em busca de estruturas de troca e parentesco Kraho, Master's thesis, Univ. of São Paulo (2016).