Epsilon Equitable Partition: A positional analysis method for large social networks

Kiran Kate

IBM Research - India
Bangalore, India
kirankate@in.ibm.com

Balaraman Ravindran

Dept. of Computer Science and Engg.
Indian Institute of Technology Madras, Chennai, India
ravi@cse.iitm.ac.in

Abstract

Positional analysis is considered an important tool in the analysis of social networks. It involves partitioning of the set of actors into subsets such that actors in a subset are similar in their structural relationships with other actors. Traditional methods of positional analysis such as structural equivalence, regular equivalence, and equitable partitions are either too strict or too generic. For real world large graphs, most of these methods result into almost trivial partitions. We propose a useful relaxation to the concept of equitable partition called an epsilon equitable partition. A variant of epsilon equitable partition called maximal epsilon equitable partition is also proposed and formulated as an optimization problem. A fast algorithm for computing epsilon equitable partitions is proposed. We also present the results of performing positional analysis on a number of networks and demonstrate empirically that positional analysis with our notion gives raise to non-trivial and meaningful partitions. Along with the static network analysis with respect to positions, we study the impact of positional analysis on the evolution of networks. Our results show that positional analysis indeed plays an important role in the evolution of networks.

1 Introduction

Positional analysis, or role analysis, is one of the important methods for analyzing the structure of a social network. The assignment of positions to actors is done based on their connectivity structure. Actors similarly embedded in the network are given the same position. The objective is to assign meaningful positions to the actors such that the structural and behavioural properties of the network can be studied with respect to the positions. The main challenge in solving a positional analysis problem is to define an appropriate notion of position which is easy to compute and which results in a meaningful interpretation.

It is natural for human beings to extract useful abstractions from the data about entities and their relationships to each other. For example, from the data about nations and their trading relationships with each other, we naturally tend to look for nations which play the roles of key importers, exporters etc. Thus, extracting such patterns from a network helps the analysis of the structure of the network at an abstract level. Assignments of positions to actors have proven to be useful for analysis of different types of networks [1, 2, 3]. It is conjectured that position plays an important role in the evolution of networks as actors belonging to a position tend to evolve similarly [4]. Positional analysis can also be treated as graph clustering and hence can have similar applications. But it is different from the traditional clustering approaches in that it considers the network connections and their properties in grouping the actors together. The applications of positional analysis also include graph summarization and enhancing graph searchability.

However, mining such roles from the network data though easy for humans, is difficult to automate. Research in the area of positional analysis has shown that coming up with useful abstractions of a network is a non-trivial problem. Various definitions of a position have been suggested but as discussed in the next section, the existing notions are either too strict or too generic to be useful in practice. Also, the positional analysis work so far mainly deals with smaller graphs and we observed that the usability of the existing methods for large complex networks is severely limited.

We propose a relaxation to an existing positional analysis method which leads to more useful abstractions. This relaxed formulation is based on the notion of *ϵ-equitable partition* and the corresponding optimized version, *maximal ϵ-equitable partition*. We also give a heuristic based algorithm to compute the partition of a graph based on our notion of position. We show that our method can be applied to real world

15^{th} International Conference on Management of Data
COMAD 2009, Mysore, India, December 9–12, 2009

large and complex graphs and explain the meaningful partitions of such graphs in the results section.

The partitioning of a graph using positional analysis is a relatively unexplored area compared to the partitioning using dense communities. The difference in the two approaches is that the communities tend to divide the actors such that actors in a subgroup are densely connected to each other [5, 6, 7]. On the other hand, positional analysis tries to find actors similarly embedded in the network. Actors in the same position can be far from each other and may not even be reachable from each other but they are similar in terms of their connectivity in the network. Such partitions help in the structural analysis of graphs. For example, actors in a particular position could all have high centrality, and vice versa. Also, analyzing the behaviour of the positions can find application in problems like anomaly detection and churn prediction. We believe that our work is the first one which tries to apply positional analysis to real world large complex networks and their evolution.

The rest of the paper is organized as follows: We present the background on the existing methods of positional analysis in Section 2. We introduce our definition of *maximal ϵ partition* and discuss the advantages and the algorithm proposed in Section 3. The evaluation of our method on a number of datasets is given in Section 4. Finally, we conclude and discuss the planned future work in Section 5.

2 Existing Methods

To understand the concept of positional analysis, let us consider the example network given in Fig. 1. It is a network showing the relationships between a teacher (node A), three teaching assistants (nodes B, C, and D) working under A and the students related to the teaching assistants (TAs). Performing positional analysis on this network would give us a partition of the nodes such that each block of the partition represents a position. Intuitively, the first level of abstraction suggests positions like teacher, TA, student. We explain some of the existing methods and their limitations with the help of this example and we also explain the result of our method on this network in section 4.

Structural equivalence. Two actors are *structurally equivalent* if they have identical ties to and from all other actors in the network [8]. Though this notion of position is widely used for small networks, it is a very strict definition and hence we rarely find structurally equivalent actors in real world networks.

Automorphism. Given a graph G = <V, E>, an *automorphism* is a bijective function f from V to V such that (a, b) $\in$ E if and only if (f(a), f(b)) $\in$ E.

Automorphism is an isomorphism from a graph to itself. The orbits of an automorphism group form a partition of the graph and each block of this partition indicates a position [10]. The problem of finding all automorphically equivalent vertices is known to be computationally hard. However, efficient graph automorphism solvers like NAutY - No Automorphisms, Yes? [9] based on McKay [10] exist which are widely used for solving this problem. Automorphism is also a strict notion for position since it is a bijective function. It fails to characterize the real world notion of similarity as real world networks are quite irregular and existence of symmetries is very rare in such networks.

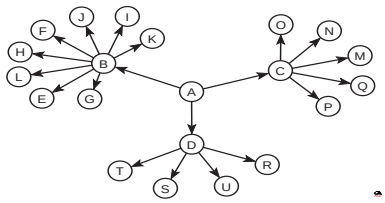

Figure 1: Example 1: A teachers - TAs - students network

Regular equivalence. A regular partition is a partition of nodes into classes such that nodes of the same class are surrounded by the same classes of nodes [11]. It is same as the notion of bisimulation from computer science [12].

Definition Given a graph G = <V, E> and $\equiv$, an equivalence relation on V, $\equiv$ is a *regular equivalence* if and only if for all a, b, c $\in$ V, a $\equiv$ b implies:

1. (a, c) $\in$ E implies there exists d $\in$ V such that (b, d) $\in$ E and d $\equiv$ c and

2. (c, a) $\in$ E implies there exists d $\in$ V such that (d, b) $\in$ E and d $\equiv$ c.

A graph may contain several regular equivalences and it is shown that the set of regular equivalences forms a lattice [13]. The supremum element of the lattice is known as the maximal regular equivalence or MRE. For undirected graphs with no isolates, the MRE is trivial and results in a complete partition which is a partition with one block containing all the nodes.

For the directed graph of Fig. 1, the maximal regular partition is given in Fig. 2. This result seems quite reasonable as the first level of abstraction, but in practice, this may not relate much to the structural properties since the degree of the nodes is completely ignored while forming the partition. In the given network, say, a TA having a large number of students under him/her would have different structural and behavioural properties than a TA having a small number of students. But maximal regular partition would still put them in the same block. As mentioned earlier, the

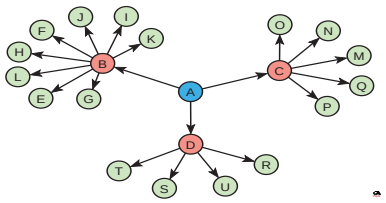

Figure 2: Regular partition of example 1. Positions are indicated by colors, thus the regular partition is {{A}, {B, C, D}, {E, F, G, H, I, J, K, L, M, O, P, Q, R, S, T, U}}

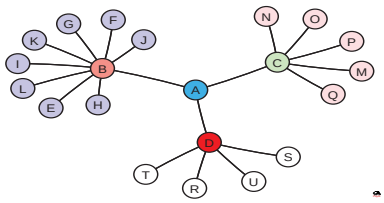

Figure 3: equitable partition of undirected version of example 1. Positions are indicated by colors, thus the equitable partition is {{A}, {B}, {C}, {D}, {E, F, G, H, I, J, K, L}, {M, N, O, P, Q}, {R, S, T, U}}

undirected version of this graph has a trivial regular partition. When the maximal regular equivalence is trivial, deciding on what element of the regular lattice should be chosen as a final regular partition is not obvious. Generation of all the elements of the lattice is computationally expensive. Two algorithms REGE and CATREGE [15] define a measure of regular equivalence between two actors but the partitions produced using these algorithms can not be characterized in terms of any graph theoretic properties and the similarity measure lacks a theoretical support.

Equitable partition. A partition Π is said to be *equitable* if,

$\forall C_i \in \Pi$, $v_1, v_2 \in C_i$ implies that $d(v_1, C_j) = d(v_2, C_j)$, $\forall C_j \in \Pi$, where

$$d(v_i, C_j) = \text{sizeof}\{\ v_k \mid (v_i, v_k) \in \text{E and } v_k \in C_j\ \}$$

Equitable partition is a relaxation of automorphism since the partitions formed by automorphism are always equitable but the reverse is not true. Polynomial time algorithms exist for finding the coarsest equitable partition of a graph [10, 16]. It can be observed that equitable partition is a regular partition with an additional constraint that the number of connections to the neighbouring positions should be equal for equivalent nodes. This constraint is too strict for complex large graphs and hence results in almost trivial partitions (with most of the blocks having small sizes). For example, for the undirected version of the graph of Fig. 1 (obtained by ignoring the directions), the coarsest equitable partition is given in Fig. 3. Nodes C and D are TAs under the same teacher, handling almost the same number of students but equitable partition put them in different blocks just because their degrees differ by one.

3 Relaxed Equitable Partitions

3.1 Problems Addressed

In previous sections, we have seen the importance of meaningful positional analysis and problems with the current methods of positional analysis. With increasing network size, it is even more important to define a notion of position which results in a meaningful partitioning of large graphs. As discussed, analysis of complex social networks using structural equivalence and automorphism tends to result into trivial partitions. For undirected graphs, even regular equivalence gives a trivial partition. Considering regular equivalence and equitable partition, we observe three major limitations:

1. Regular equivalence does not take the number of connections to other positions into account. For example, node a_1 having 10 connections to position p_1 is considered equivalent to node a_2 having just 1 connection to position p_1. This may not be suitable for most of the analysis problems.

2. Regular equivalence, however, is strict when comparing two actors based on the positions in the neighbourhood. For example, if node a_1 has a single connection to position p_1 and node a_2 does not have any connection to position p_1, a_1 and a_2 are not equivalent according to the definition, but again, for complex networks, this restriction leads to trivial partitions.

3. Definition of equitable partition rectifies the limitation 1 listed above, but it imposes a strict condition that the number of connections to other positions should be exactly equal for two nodes to be equivalent. For example, if node a_1 has 10 connections to position p_1, the node a_2 should also have 10 connections to position p_1 for it to be equivalent to a_1.

3.2 Maximal ϵ-Equitable Partition

The proposed solution is a useful relaxation to the definition of equitable partition with some added constraints. Before defining the term *maximal ϵ-equitable partition*, we define a few notations and terms as follows:

Definition Given a graph G = <V, E> and a partition $\Pi = \{C_1, C_2, ..., C_K\}$,

$$d(v_i, C_k) = \text{sizeof}\{\ v_j \mid (v_i, v_j) \in \text{E and } v_j \in C_k\ \}$$

The term $d(v_i, C_k)$ is thus the number of vertices in C_k which are adjacent to v_i. We know that the block C_k denotes a position and thus the $d(v_i, C_k)$ is the number of connections actor v_i has to the position denoted by the block C_k.

Definition Given a graph G = <V, E> and a partition $\Pi = \{C_1, C_2, ..., C_{K_\Pi}\}$, *slack* of a node v_i is defined as,

$slack_i^\Pi = ||\frac{1}{sizeof(C_{v_i})-1} \sum_{v_j \in C_{v_i} i \neq j} (\vec{\epsilon} - \mid \vec{d_i} - \vec{d_j} \mid)||$, where

$\vec{d_i}$ = the degree vector of node v_i such that $d_{i_k} = d(v_i, C_k)$, for $k = 1, 2, ..., K_\Pi$.

C_{v_i} = the block to which node v_i belongs

$\vec{\epsilon} = K_\Pi$ dimensional vector such that $\epsilon_k = \epsilon$, for $k = 1, 2, ..., K_\Pi$, where ϵ is an integer greater than zero.

$||\ ||$ is the l_1 norm of a vector (sum of the components)

Slack of a node denotes how close it is to the other nodes in the block. Large value of *slack* indicates small within-block distance. We give a new definition of partition such that each block indicates a position as follows:

Definition Given a graph G = <V, E>, partition $\Pi = \{C_1, C_2, ..., C_{K_\Pi}\}$ is *maximal ϵ-equitable* if,

1. $\forall C_i \in \Pi$, $v_1, v_2 \in C_i$ implies that $\mid d(v_1, C_j) - d(v_2, C_j) \mid <= \epsilon$, $\forall C_j \in \Pi$
2. $\forall \Pi'$ satisfying condition 1, $(K_\Pi - \sum_{v_i} slack_i^\Pi) \leq (K_{\Pi'} - \sum_{v_i} slack_i^{\Pi'})$, $i = 1, 2, ..., |V|$

The definition can be explained as follows: The first condition of the definition proposes a relaxation to the strict condition of equitable partition. We are allowing an error of ϵ in the number of connections to positions for two actors to be equivalent. For example, if ϵ is 2 and node a_1 has 10 connections to position p_1 then it is enough for node a_2 to have more than 7 and less than 13 connections to position p_1 for it to be equivalent to a_1. If Π is such that condition 1 alone holds, then Π is an *ϵ-equitable partition*.

The second condition of the definition is an optimization condition which implies that the number of blocks in the partition should be minimum and the sum of the slacks of all the nodes should be maximum. Thus we are looking for a coarsest partition satisfying the ϵ condition such that the sum of the within block distance for all the blocks is minimum.

3.3 Advantages

1. The proposed relaxation is useful in large social networks where not many actors are equitable. Say, with ϵ value of 2, in IMDB [17] graph, an actor who has worked with 10 directors, 20 actresses would be equivalent to an actor who has worked with 12 directors and 18 actresses which is fine since the real world notion of position is not that strict.

2. Also, the parameter ϵ in the definition lets us tune the amount of relaxation. The special case of the definition is ϵ value of 0 which corresponds to a coarsest equitable partition.

3. It can be seen that the first condition indicates a stricter version of regular equivalence with respect to the number of connections to positions (as regular equivalence does not care about the number of connections as long as there are some). It also solves the problem of strictness of regular equivalence, since a node a_1 having no connections to position p_1 and node a_2 having ϵ connections to position p_1 can belong to the same block of the *maximal ϵ-equitable partition*. With an ϵ value of *infinity* and a condition that non-zero values of $d(v_i, C_k)$ are not to be treated ϵ-equal to zero values, the definition results in a maximal regular partition.

4. *Maximal ϵ-equitable partition* is an intuitively appealing notion of equivalence. We can give a guarantee on the amount of error each block would have in terms of the slack.

3.4 Computation Issues

It is known that clustering a set of data with a given number of clusters to minimize the mean squared distance from each data point to its nearest center is an NP-hard problem [18] and approximation algorithms exist for the same. The proposed formulation of *maximal ϵ-equitable partition*, intuitively, is harder than those kinds of problems since we are not assuming the value of K to be known. Roberts and Sheng [19] show that the optimization approaches to regular equivalence are NP-complete. Hence, based on the known problems of similar kind, we can safely assume that the problem of finding a *maximal ϵ-equitable partition* of a graph is hard.

The most naive solution to the problem is to exhaustively try all possible partitions and pick the one which satisfies the proposed conditions. This approach clearly leads to an exponential algorithm as the number of possible partitions is $\sum_{k=1}^{n} k^n$, n is the number of nodes in the graph. Ideally, we should come up with a polynomial time approximation algorithm with a bounded approximation ratio. Currently, we propose

two approaches for obtaining the proposed partition from a graph as follows:

1. Top-down approach: This approach starts with a complete partition (a partition with a single block containing all the nodes) and works iteratively to split the existing blocks into two or more in each iteration. We tried a number of greedy heuristics with this approach two of which gave meaningful and almost optimal partitions for small networks. But they turned out to be computationally expensive to be applied to large graphs. We do not report those methods here as the main objective is to present a scalable method.

2. Bottom-up approach: This approach starts with an equitable partition of the graph and iteratively merges two or more blocks in each iteration. We tried some greedy heuristics with this approach too and we report the most efficient of them.

3.5 Our Algorithm

Algorithm 1 An efficient algorithm to find ϵ-equitable partition

1: Sort the input equitable partition according to ascending order of the degree of the blocks (degree of the block of an equitable partition is same as the degree of the member nodes of that block)
2: **for** i = 0 to ϵ **do**
3: merge all the blocks having degree = i into a single block and update the partition by deleting the original blocks and by adding the new block update the variable numBlocks according to the resulting partition
4: **end for**
5: **for** each node of the graph **do**
6: calculate the degree vector $\vec{d_i}$
7: **end for**
currentBlock = the first block in the ordered partition having degree $> \epsilon$
8: **for** each block in the current partition **do**
9: check if it can be merged with currentBlock without violating the ϵ criterion, where $\epsilon = \epsilon/2$ if yes, merge it with currentBlock and update the partition, numBlocks, and the degree vectors else make the block as currentBlock and continue
10: **end for**

As mentioned before, we tried a number of heuristic based algorithms for computation of *maximal ϵ-equitable partition* but most of them were computationally intensive. The algorithm explained in this section aims at scalability and hence tries to follow the bottom-up approach with minimum processing required to choose the candidates to merge. The algorithm is a simple single iteration trying to merge the blocks of the coarsest equitable partition of the graph. The hierarchical approaches tried were expensive and since the goal of this work is to demonstrate positional analysis of large graphs, we present the simplest yet efficient algorithm.

A coarsest equitable partition of the given graph is obtained using the tool NAutY [9]. Nauty finds equitable partition of a graph as the first step towards finding its automorphisms [10]. It has an efficient implementation of an iterative algorithm to find the coarsest equitable partition of a graph. The time complexity of this algorithm is reported as $O(n^2 \log n)$ where n is the number of nodes in the graph.

The input to our algorithm is the original graph, the coarsest equitable partition of the graph and a value for ϵ. The algorithm proposed tries to merge the blocks of the equitable partition such that the merging does not violate the ϵ criterion defined by *maximal ϵ-equitable partition*. There are a few mathematical observations which form the basis of the merging step of the algorithm which are given as follows.

Observation 1: Merging the blocks of an equitable partition having same degree such that it is less than ϵ does not violate the ϵ condition defined by *ϵ-equitable partition*. This mathematical property can be proved as follows:

For all the elements with degree less than ϵ, all the components of the degree vector (as defined in the algorithm) are less than ϵ. It can be easily seen that the difference of two numbers which are less than ϵ is never greater than ϵ. This shows that the block obtained by merging the elements with degree less than ϵ would never violate the ϵ criterion. Since this merging is to be performed on equitable partition, the members of rest of the blocks have identical degree vectors before merging and hence there is no violation after merging.

Observation 2: Merging blocks according to the ϵ criterion with $\epsilon_1 = \epsilon/2$ does not violate the ϵ condition for other blocks. It can be proved as follows:

Consider the partition {A, B, C} such that the last block either corresponds to a block of the initial equitable partition or is obtained by merging blocks with $\epsilon_1 = \epsilon/2$. Say, we merge the blocks A and B with $\epsilon_1 = \epsilon/2$. Before merging, the degree vectors of any two arbitrary nodes c and d from block C were
$\vec{d_c} = (d_{c1}, d_{c2}, d_{c3})$
$\vec{d_d} = (d_{d1}, d_{d2}, d_{d3})$
where the positions 1, 2, and 3 correspond to blocks A, B, and C respectively. We claim that, after merging of blocks A and B, the block C would not violate the ϵ condition. To see this, lets write down the degree vectors for node c and d after the merger in terms of the quantities before the merger.
$\vec{d_c} = (d_{c1} + d_{c2}, d_{c3})$

$\vec{d_d} = (d_{d1} + d_{d2}, d_{d3})$
We want to show that $|(d_{c1} + d_{c2}) - (d_{d1} + d_{d2})| < \epsilon$. Since we said that the block C satisfies the ϵ condition with $\epsilon_1 = \epsilon/2$,

$$|d_{c1} - d_{d1}| < \epsilon/2 \tag{1}$$

$$|d_{c2} - d_{d2}| < \epsilon/2 \tag{2}$$

Now, summing up the above two equations,

$$|d_{c1} - d_{d1}| + |d_{c2} - d_{d2}| < \epsilon \tag{3}$$

We know that for any two real numbers x and y, $|x + y| < |x| + |y|$, hence

$$|d_{c1} - d_{d1} + d_{c2} - d_{d2}| < \epsilon \tag{4}$$

$$|(d_{c1} + d_{c2}) - (d_{d1} + d_{d2})| < \epsilon \tag{5}$$

Hence the result. During any stage of the execution of our algorithm, when two blocks are to be merged, each block from the set of the rest of the blocks is either same as one from the equitable partition or was obtained by merging in some previous iteration using the updated value of ϵ. So even if the above discussion considers only 3 blocks, since the blocks are independent, it applies to any number of blocks. This suffices to prove that the algorithm proposed finally results in an *ϵ-equitable partition* with the value of ϵ given by the user.

The algorithm based on these two observations is given in Algorithm 1. The time complexity of the algorithm is $O(n^3)$ where n is the number of nodes in the graph. Step 1 of the algorithm takes $O(n \log n)$ time for sorting. The *for* loop of line 2 takes $O(n)$ time. Calculation of degree vectors for all nodes for the first time involves visiting all the edges twice and hence takes $O(n^2)$ time. The *for* loop of line 9 executes for each block of the partition and there could be a maximum of n blocks in a partition. The body of the loop involves traversing the degree vectors of all the nodes of currentBlock, each vector can be of a maximum length of n (the maximum number of blocks) and there could be a maximum of n nodes in the block. The updation of degree vector after merging of the blocks can be done in constant time by just adding the components corresponding to the merged blocks. The entire *for* loop of line 9 thus takes $O(n^3)$ time. However, it should be noted that, this is a very loose theoretical upper bound, in practice, the algorithm is very efficient since the number of blocks in the partition and the average number of nodes in a block are inversely proportional to each other. The algorithm to obtain equitable partition is reported to have time complexity of $O(n^2 \log n)$. Therefore, the overall complexity of computing the *ϵ-equitable partition* is $O(n^3)$.

4 Experimental Results

In this section we discuss the results of our method on a number of datasets. We demonstrate that our method gives meaningful and non-trivial abstractions facilitating structural and behavioural analysis of the graphs. We compare it against equitable partitions on different parameters and show that it is more useful and still is quite scalable to large graphs. We also conduct experiments observing the evolution of the networks and study the role of positions in the evolution. We compare the results of our method with the preferential attachment model [20] to show that positions form a more stable and stronger basis for evolution than just the degree of the nodes.

4.1 Datasets

A kinship network [21] that consists of a set of sixteen Italian families in the early 15^{th} century was considered for the preliminary analysis. The relation modelled in the network is that of marriage between pairs of families. The nodes are labelled with the surnames of the families and there are 20 edges indicating marriages between the pairs of actors. This dataset was used for manual validation of the results since it is a real life social network of small size.

IMDB [17] dataset was used to construct a network of actresses of Hindi movies. Hindi was chosen as a language to use the familiarity with the domain to facilitate intuitive evaluation. An edge between two actresses in the graph indicates that they worked together for one or more movies. The network was created for years 2000 to 2009 such that it includes all the movies released on or before that year. The sizes of these networks range between 3165 and 5328 nodes (37716 and 60947 edges).

4.2 Static Analysis

In this section, we take 3 case studies of networks of different sizes and evaluate their partitions intuitively. Such intuitive explanation of the results is common in the work related to graph partitioning since there is no labelled data available for evaluation of such tasks [22, 23]. We also evaluate the results of partitioning the IMDB network quantitatively on some parameters and compare it with existing methods.

4.2.1 Network of Example 1

For the undirected version of the graph of Fig. 1, the *maximal ϵ-equitable partition* with ϵ value of 1 is given in Fig. 4. It can be seen that the result is a meaningful abstraction of the graph. It not only identifies teacher, TAs and students correctly, but also treats nodes C and D as equivalent since they are TAs under the same teacher, handling almost the same number of students. It also differentiates B from them since it is connected to more number of students and hence

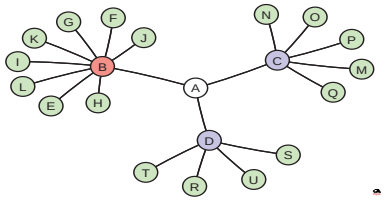

Figure 4: *maximal ϵ-equitable partition* of undirected version of example 1 with $\epsilon = 1$. Positions are indicated by colors, thus the *maximal ϵ-equitable partition* is {{A}, {B}, {C, D}, {E, F, G, H, I, J, K, L, M, O, P, Q, R, S, T, U}}

might have different properties than C and D. The value of ϵ should be chosen according to the perception of position for the graph. Studying the structural properties of the graph like average degree, mode of degree etc. can help decide the value of ϵ.

4.2.2 Kinship Network

The quality of the results of our method can be judged by manual inspection of the partition for small networks. Figure 5 shows the result for the Italian families kinship network with $\epsilon = 2$. Nodes belonging to the same position are colored same. The regular partition of the kinship graph is a complete partition and the equitable partition is a discrete partition with 16 blocks. The result can be explained as:

Pucci is an isolated node with no relations with others and hence represents the isolate position. On the other hand, Medici is the most central node with a degree more than two and half times the average degree of the graph and hence occupies the central position. The families Pazzi, Acciaiuoli, Ginori, and Lamberteschi have had relationships with one family each. Since these nodes have just one relationship each, for most of the problems we are interested in, it does not matter much with whom they have that connection. Similar is the case with Salviati and Barbadori but they still have connection to Medici in common.

Ridolfi and Tornabuoni are regular equivalent to each other with the same number of connections to the neighboring positions, Albizzi is similar with one different connection. Castellani and Bischeri however, have 2 and 3 connections respectively to the last but one position unlike the others in their block (others have just one connection to that position). But they are still treated the same since the allowed relaxation is of 2. Peruzzi and Strozzi are regular equivalent with a difference of 1 in connections to one of the neighboring positions. Inclusion of Guadagni follows due to the allowed relaxation.

It can be observed that the number of blocks is much less than those of the equitable partition. The usability and meaning of the blocks depends on the choice of ϵ which in turn depends on the user's desired level of abstraction and approximation.

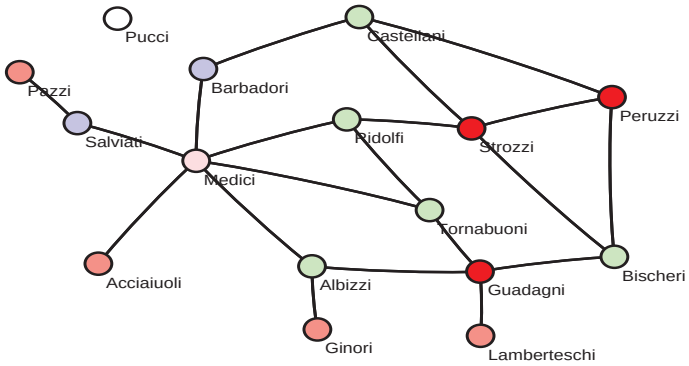

Figure 5: Result of our method on kinship dataset with $\epsilon = 2$. The partition is {{Pucci}, {Acciaiuoli, Pazzi, Ginori, Lamberteschi}, {Salviati, Barbadori}, {Albizzi, Tornabuoni, Ridolfi, Castellani, Bischeri} {Peruzzi, Guadagni, Strozzi}, {Medici}}

4.2.3 IMDB Network

Intuitive Evaluation

We focus the intuitive evaluation of the results of our method on analysis of IMDB Hindi actresses graph. To demonstrate the quality of the partition, we discuss a few positions and their characteristics. The actresses graph was created for year 2009 which means all the movies released on or before 2009 are considered for creation of the network. Since this is a dense graph with 5328 nodes and 60947 edges, the value of ϵ is chosen to be 10 for this analysis.

The *ϵ-equitable partition* has 770 blocks, giving an approximate average block size of 7. As given later in the paper, the blocks are found to be homogeneous with respect to the number of movies the member actresses worked for. We also observe many interesting blocks which could be characterized in some way or the other. We present some of those cases here starting with the nodes having higher degrees.

Actresses Anjana Mumtaz and Shashikala are found to be in one block who share the property of being famous actresses who started with small roles in the movies and then evolved as famous mothers of the lead roles. Famous actresses in lead role Amrita Singh and Meenakshi Sheshadri who started their career in the same year (1983) and have shown similar career graph with respect to the number of movies, types of movies, types of roles, and popularity, form a block of the partition.

It should be noted that the nodes with high values of degree do not form large size blocks but those with smaller degrees tend to be part of larger blocks. One of the blocks is {Sonu Walia, Bharati Achrekar, Shefali, Anita Guha, Kimi Katkar} where all the actresses have acted in supporting roles and two of them have been considered for one of the famous awards for best supporting role. Another similar block {Simone

Singh, Tanaaz Currim, Ranjana Sachdev, Deepshika, Neeta Mehta, Vidya Sinha, Padma Rani} corresponds to relatively younger actresses in supporting role who have also acted on TV. One block consists of actresses Rajshree, Babita Kapoor, Nagma, and Rambha which happen to be the actresses in lead role who gained popularity even though they have acted in very few movies.

Note that, while most of the blocks contain actresses who worked in a similar time period, some blocks show a mix of old and new actresses who share some other common characteristics. As most of the popular actresses have high nodal degrees in this graph, they tend to be singletons. As we move on towards lower degrees, we find a number of large blocks which contain almost all the actresses of a particular movie/TV show. The examples are: a block of 37 actresses who acted as models for movie 'Fashion (2008)', block of 31 actresses who acted as players in the movie 'Chak de India', block of 24 actresses from the movie 'Bend it like Beckham', block of 10 actresses from the famous TV series 'Kyunki... Saas Bhi Kabhi Bahu Thi...' etc.

Thus the analysis shows that the positions do correspond to the real positions in many cases. Some of the blocks do not show any known characteristics but the analysis is limited due to unavailability of extra information about those actresses. Some cases might not even have any common characteristics in terms of actor attributes but their structural characteristics still follow from the definition of the partition. This is expected since we know that a dataset can be clustered naturally in multiple ways and combining our approach with other attributes and methods can perform even better in such cases.

Quantitative Evaluation

Validation of positional analysis results using actor attributes is one of the known techniques of evaluation [21]. In the network of actresses of Hindi movies, we consider the number of movies as the attribute for validation. We expect the actresses in a position to have acted in almost the same number of movies. Table 1 gives the results of such analysis. The year column shows the year the graph was created for, as explained earlier. We calculated the standard deviation of the number of movies for all the blocks of the partition. The average stddev column shows the average of the standard deviation of the blocks.

The degree distribution of IMDB follows power law as expected [24] and by intuition we can expect the number of movies to be correlated to the number of co-actresses. Hence the number of movies distribution for the nodes can be expected to follow power law. Since the blocks of the partition are more or less homogeneous with respect to degree, we might attribute the homogeneity with respect to movies to the degree. To show that degree alone is not a good reason for blocks to have actresses who have acted in almost same number of movies we report the results for the partition based on just the degree of the nodes. The row having no value for ϵ indicates such a partition.

Table 1 shows that the partitions obtained using our method are reasonable with respect to the number of blocks compared to the equitable partition. As discussed, equitable partitions of such real world large graphs turn out to be almost trivial with the average size of the blocks ranging from 1 to 2. However, our relaxation results in non trivial partitions which could be followed for further analysis of the positions.

The average stddev column shows that the blocks in *ϵ-equitable partitions* are quite homogeneous with respect to the number of movies. The homogeneity with respect to number of movies decreases with increase in the value of ϵ as expected but the average stddev is still less than 1 for almost all the cases. Though we observe that the stddev has very small values for most of the blocks compared to the average, we report the average just to give an idea of the values. The average mean values for the number of movies of the blocks were almost in the same range for all the partitions and hence are not reported here.

Equitable partition consists of a large number of singleton blocks and a number of blocks of small sizes and hence report zero or very low value of the standard deviation for most of the blocks. The high value of the average stddev for the partition based on nodal degrees clearly shows that degree is not a sufficient criterion for such analysis.

Table 1: Statistics of the positions of IMDB network with respect to the number of movies. * - The row corresponds to a partition based only on degree.

Year	ϵ	No. of nodes	No. of blocks	Average stddev of no. of movies
2000	0	3165	2233	0.0022
2000	2	3165	1218	0.2433
2000	4	3165	852	0.6138
2000	6	3165	617	1.0850
2000	-*	3165	206	4.3302
2009	0	5328	3201	0.0024
2009	2	5328	1871	0.1531
2009	4	5328	1423	0.3752

4.3 Dynamic Analysis

We study the evolution of the IMDB Hindi actresses network with respect to the positions. It is interesting to find that, to a large extent, social networks evolve according to the positions. The procedure followed for such analysis is as follows: We construct the IMDB Hindi actresses network for years 2000 to 2009. We perform positional analysis on these 10 networks. The positions of a network are compared with the positions obtained for another network such that the second network is an evolved version of the first network. For

example, the partition of the network corresponding to year 2000 is compared with the partitions of networks corresponding to years 2001 to 2009. For each position of the first network we find a corresponding position in the second network such that the size of the intersection (the set intersection) of these positions is maximum possible. We consider the second position found in such a way to have evolved from the first position if the number of nodes in their intersection is greater than or equal to α percentage of the number of nodes in the first position.

The procedure can be understood better with an example. Assume that the networks under study are networks for year 2000 and 2001 and the value of α is chosen to be 90. For a particular position of the year 2000 network, say p_1, we iterate over all the positions of year 2001 network to find a position say p_2, such that the size of intersection is maximum possible. We say that position p_1 has evolved into position p_2 if the size of intersection of p_1 and p_2 is greater than or equal to 90% of the size of p_1. In such a case, the nodes belonging to the intersection are considered to follow a position. We add up the number of nodes from year 2000 network which follow positions while evolving into year 2001 network and report their percentage relative to the size of the network (the total number of nodes).

Positional analysis is performed using equitable partition and ϵ-*equitable partition* methods. During our study we observed that, to a large extent, positions given by equitable partitions evolve similarly. One reason for this is the large number of singleton blocks in equitable partitions. The other reason could be that since the blocks of an equitable partition have same degrees, according to the preferential attachment model [20], nodes with same degree tend to evolve similarly. The probability of a new node attaching to an old node is directly proportional to the degree of the old node. In order to study the effect of this, we partition the graph based on the degrees of the nodes such that same degree nodes are put in the same block.

Tables 2, 3, and 4 summarize the results of such analysis for different values of α. The source network indicates the base network for evolution. The second column indicates the value of ϵ, 0 indicates equitable partition and '-' indicates a partition based on nodal degree. All the other values are the percentages of nodes following positions during the evolution from the network given in that row to network given in that column for the method indicated by corresponding value in second column. Figures 6, 7, and 8 facilitate the comparison of different methods for the evolution of the year 2000 network.

Table 2 and Fig. 6 indicate that equitable partitions perform the best during the evolution as the percentage of nodes following positions is consistently high for all the years with an average of around 88%. But we know that, equitable partitions are almost trivial and hence have many singleton blocks. Also, the degree of the nodes in a block is same in an equitable partition. These two factors lead to the high values of the percentages. Our method performs consistently with an approximate average of 47%. The partition based on nodal degree however performs very poor with an average of around 4.5%.

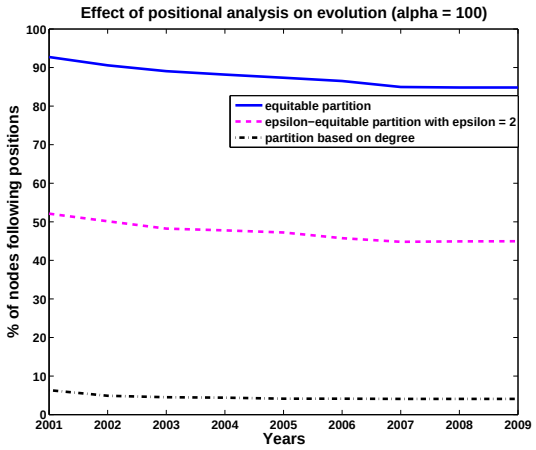

Figure 6: Comparison of the % of nodes following positions given by different partitioning methods in the evolution of IMDB Hindi actresses year 2000 network. $\alpha = 100$

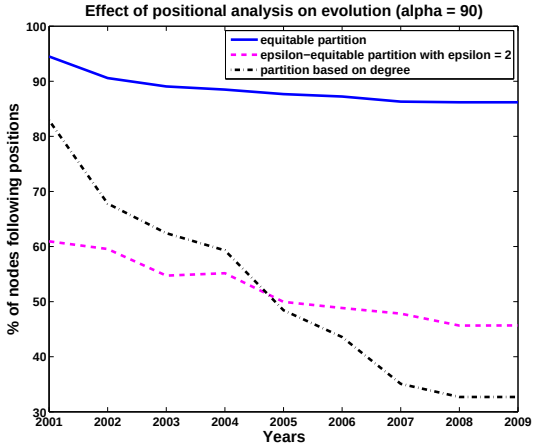

Figure 7: Comparison of the % of nodes following positions given by different partitioning methods in the evolution of IMDB Hindi actresses year 2000 network. $\alpha = 90$

Table 3 and Fig. 7 also show similar trend for equitable partitions. But since the value of α is smaller, more number of positions were qualified according to our criterion, resulting in higher percentages of nodes. Our method also shows similar trend with a significant

Table 2: Percentage of nodes following positions during evolution for different positional analysis techniques. α = 100

Source Network	ϵ	2001	2002	2003	2004	2005	2006	2007	2008	2009
2000	0	92.73	90.58	89.06	88.18	87.36	86.50	84.96	84.83	84.83
2001	0		93.90	91.62	89.74	88.04	87.17	86.30	86.00	86.00
2000	2	52.10	50.14	48.24	47.77	47.23	45.75	44.80	44.92	44.96
2001	2		53.27	50.16	49.59	47.65	46.90	46.33	46.00	45.65
2000	-	6.35	4.89	4.51	4.39	4.13	4.13	4.07	4.07	4.07
2001	-		8.63	7.59	6.39	6.03	4.06	4.00	4.00	4.00

Table 3: Percentage of nodes following positions during evolution for different positional analysis techniques. α = 90

Source Network	ϵ	2001	2002	2003	2004	2005	2006	2007	2008	2009
2000	0	94.50	90.58	89.06	88.49	87.67	87.23	86.31	86.19	86.19
2001	0		96.14	92.22	91.36	90.40	89.35	87.74	87.44	87.44
2000	2	60.94	59.55	54.72	55.16	49.95	48.84	47.83	45.65	45.68
2001	2		64.96	57.75	57.54	51.74	50.79	50.13	47.44	47.08
2000	-	82.90	67.80	62.43	59.30	48.43	43.57	35.07	32.70	32.70
2001	-		84.30	66.48	63.10	57.87	55.72	54.23	41.37	41.37

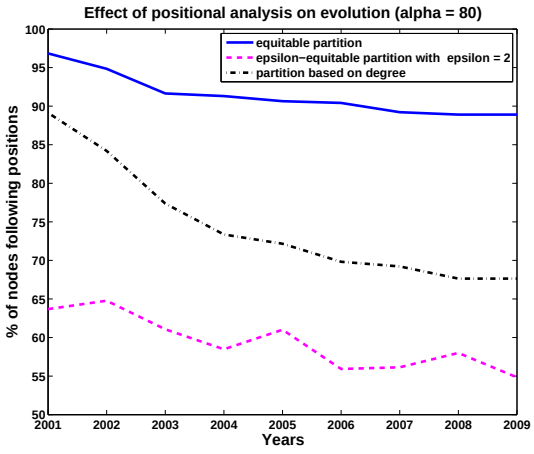

Figure 8: Comparison of the % of nodes following positions given by different partitioning methods in the evolution of IMDB Hindi actresses year 2000 network. α = 80

increase in individual percentage values. The behavior of the partitions based on degree is however very unstable. The percentage of nodes is quite high (around 82%) for time duration of one year (from 2000 to 2001 and from 2001 to 2002) and it suddenly falls to around 66% for the time gap of two years and so on, with an average value of 35% for time duration of 9 years. Such an unstable behavior can not be used as a basis of any prediction.

Similar observations can be made from Table 4 and Fig. 8. The performance of partition based on nodal degree is unstable in this case too. We can see that the slopes of the curves for our method and equitable partition are almost the same indicating that actors in a position evolve together consistently. The steep curve of the degree based partition indicates that nodes of same degree evolve similarly over a shorter period and show a large difference in the behaviour if observed for a long period. Hence degree based partition does not form a good basis for such evolution studies.

In summary, equitable partitions perform the best due to the reasons given earlier. Our method however performs consistently and beats the partitions based on degrees for higher values of α and in stable performance. The analysis clearly shows that the degree of nodes is not the only factor responsible for their behavior during evolution and positions of the nodes definitely play a major role in that process. However, finding a suitable positional analysis method for study of evolution is a challenging problem. Though equitable partitions perform well, they are of little help

Table 4: Percentage of nodes following positions during evolution for different positional analysis techniques. α = 80

Source Network	ϵ	2001	2002	2003	2004	2005	2006	2007	2008	2009
2000	0	96.84	94.84	91.65	91.31	90.64	90.42	89.22	88.90	88.90
2001	0		97.10	94.31	93.48	92.67	91.39	89.98	89.93	89.93
2000	2	63.69	64.77	61.07	58.48	61.01	55.92	56.14	58.00	54.84
2001	2		67.68	62.78	60.41	60.56	57.19	57.57	57.72	55.45
2000	-	89.13	84.20	77.37	73.36	72.16	69.82	69.22	67.64	67.64
2001	-		90.82	83.88	82.12	79.46	75.87	75.03	73.48	73.48

in practice due to the large number of blocks. Our method performs poorer when compared to equitable partitions but as showed before, leads to a reasonable number of blocks. It also is consistent unlike the partitions based on degree. Hence, our definition of a partition can be considered a promising approach for solving this problem. We believe that the performance of our method can be improved by designing a hierarchical algorithm which preserves the degrees to make the positional analysis more suitable for studying the evolution of the networks.

5 Conclusion and Future Work

The main objective of our work was to come up with an appropriate positional analysis technique for large real world networks. The proposed concept of *maximal ϵ-equitable partition* achieves this objective to a great extent and has been shown to be a significant first step in that direction. It is a useful relaxation of the concept of equitable partition in that it reduces the size of the partitions and still gives meaningful partitions. We studied and concluded that the existing positional analysis methods are not of much practical use for large complex graphs. Our method succeeds in overcoming some of the limitations of the existing approaches. The heuristic based algorithm proposed to compute *ϵ-equitable partition* is scalable and hence can be applied to large graphs to obtain partitions of good quality.

Another important conclusion of our work is that positional analysis can be effectively used in analyzing the structural and behavioral properties of the nodes of real world large networks. We showed that positional analysis is indeed important in studying the evolution of networks. According to our knowledge, no analysis similar to the dynamic analysis done by us to study the effect of positions on evolution of the network has been reported anywhere in the literature.

The results of the proposed method depend on the value of ϵ and hence it should be chosen carefully. Also, ϵ remains the same for all the nodes of the graph irrespective of their degrees. But sometimes, a particular value of ϵ might be too big when merging nodes with smaller degree while appropriate for nodes with large degrees. The future work involves deciding on appropriate schemes for choosing and applying the value of ϵ. Coming up with better algorithms, that can find guaranteed approximations to maximal epsilon equitable partitions is also one of the important next steps. We evaluated our method on real world graphs and showed intuitively that the positions make sense, but it would be worthwhile to find a labelled graph, where the nodes are marked with real world positions for evaluation. The positions found by our method can be compared against these real world positions in such case and a better validation could be obtained. The problem in this kind of evaluation though, is the unavailability of such labelled network.

The definition *maximal ϵ-equitable partition* as discussed in the paper applies to undirected simple graphs. Since real world networks can be directed, weighted, and multirelational, we plan to extend our definition to these classes of graphs.

References

[1] Johnson, J.C., Borgatti, S.P., Luczkovich, J.J., Everett, M.G.: Network role analysis in the study of food webs: An application of regular role coloration. The Journal of Social Structure **2**(3) (2001)

[2] Luczkovich, J.J., Borgatti, S.P., Johnson, J.C., Everett, M.G.: Defining and measuring trophic role similarity in food webs using regular equivalence. Journal of Theoretical Biology **220**(3) (2003) 303 – 321

[3] Welser, H.T., Gleave, E., Fisher, D., Smith, M.: Visualizing the signatures of social roles in online discussion groups. The Journal of Social Structure **8**(2) (2007)

[4] Doreian, P.: Actor network utilities and network evolution. Social Networks **28** (2006) 137–164

[5] Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. PROC.NATL.ACAD.SCI.USA **99** (2002) 7821

[6] Cai, D., Shao, Z., He, X., Yan, X., Han, J.: Community mining from multi-relational networks. In: Proceedings of the 2005 European onference on Principles and Practice of Knowledge Discovery in Databases (PKDD'05). (2005)

[7] Pandit, V., Modani, N., Mukherjea, S., Nanavati, A., Roy, S., Agarwal, A.: Extracting dense communities from telecom call graphs. Communication Systems Software and Middleware and Workshops, 2008. COMSWARE 2008. 3rd International Conference on (Jan. 2008) 82–89

[8] Lorrain, F., White, H.: Structural equivalence of individuals in social networks. Journal of Mathematical Sociology **1** (1971) 67–80

[9] McKay, B.: Nauty http://cs.anu.edu.au/bdm/nauty/

[10] McKay, B.D.: Practical graph isomorphism. Congressus Numerantium **30** (1981) 45–87

[11] White, D.R., Reitz, K.: Graph and semigroup homomorphisms on semigroups of relations. Social Networks **5** (1983) 193–234

[12] Marx, M., Masuch, M.: Regular equivalence and dynamic logic. Social Networks **25** 51–65

[13] Borgatti, S.P., Everett, M.G.: The class of all regular equivalences: algebraic structure and computation. Social Networks **11** (1989) 65–88

[14] Batagelj, V., D.P., Ferligoj, A.: An optimisation approach to regular equivalence. Social Networks **14** (1992) 121–135

[15] Borgatti, S.P., Everett, M.G.: Two algorithms for computing regular equivalence. Social Networks **15** (1993) 361–376

[16] Boldi, P., Lonati, V., Santini, M., Vigna, S.: Graph fibrations, graph isomorphism, and pagerank. RAIRO Inform. Theor. **40** 227–253

[17] Internet movie database: http://www.imdb.com

[18] Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C.D., Silverman, R., Wu, A.Y.: A local search approximation algorithm for k-means clustering. In: SCG '02: Proceedings of the eighteenth annual symposium on Computational geometry, New York, NY, USA, ACM (2002) 10–18

[19] Roberts, F., Sheng, L.: Role assignments. Combinatorics, Graph Theory and Algorithms **2**

[20] Barabasi, A.L., Albert, R.: Emergence of scaling in random networks. Science **286** 509–512

[21] Wasserman, S., Faust, K.: Social Network Analysis: methods and applications. Cambridge University Press (1994)

[22] Dhillon, I., Guan, Y., Kulis, B.: A fast kernel-based multilevel algorithm for graph clustering. In: KDD '05: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, New York, NY, USA, ACM (2005) 629–634

[23] Papadimitriou, S., Sun, J., Faloutsos, C., Yu, P.S.: Hierarchical, parameter-free community discovery. In: ECML PKDD '08: Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases - Part II, Berlin, Heidelberg, Springer-Verlag (2008) 170–187

[24] Barabasi, A.L., Bonabeau, E.: Scale-free networks. Scientific American **288** 60–69