Chapter 3

Ego Network Based Community Detection

This chapter investigates the role of nodes in community formation of real networks from the perspective of ego network and explores the ways of deepening the involvement of nodes in the community detection process.

3.1 Introduction

A fundamental problem in community detection is identification of inaccurate communities. The properties of network elements are generally explored in three levels of abstraction to correctly detect communities. Firstly, the node level properties that are associated with ground level entities of the network such as nodes or connections. The node level properties such as various node centrality measures [61, 97, 147, 189] and similarity between two nodes [74, 135, 177] are used extensively for community detection. Secondly, the group level or community level properties that are associated with group of nodes or

connections or sub-graphs. Popularly used community level properties include modularity of community [143], similarity of communities [50] and density of communities [169]. Lastly, the network level properties that deal with various properties of the network. Network level properties are defined mainly in terms of cut, which include network level properties such as ratio cut [110, 208], normalized cut [178] and conductance [21] etc.

Community detection algorithms utilize one or more of the properties discussed above to identify communities [54, 169, 200]. However, algorithms mostly incorporate these properties only to determine community members skipping completely the viewpoint of nodes that whether the node is likely to join the community or not. Thus, the role of individual nodes in existing algorithms is limited as the membership of a node is decided from the viewpoint of other nodes not the node itself. ENBC algorithm is proposed to enhance the role of individual nodes in community detection process. A person centric network called ego network is studied to introduce the notion of mutual interest (see section 3.1.2), where membership of a node is examined from the viewpoint of other nodes as well as the node itself. ENBC algorithm utilizes the notion of mutual interest. Involvement of individual nodes in ENBC algorithm show highly accurate communities.

3.1.1 Ego Network

Ego network is a well known social phenomenon which deals with personal interest and the relationship [11, 55, 129]. Virtual network built around any arbitrary person as shown in the Figure 3.1, accompanying those persons with whom the person has direct relationship is called ego network. The person with respect to whom the network is drawn referred as ego and persons connected with the ego are termed as alters. Connections between ego and alters in the network are referred as ties. Different levels of ego network are defined based on the distance from ego to alters.

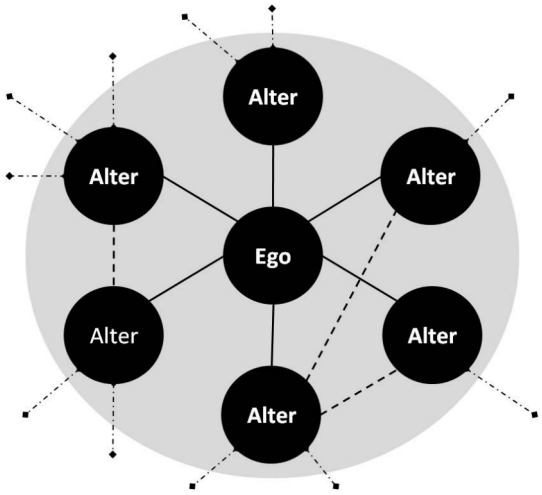

Figure 3.1: Example of $\eta^{1.0}$ ego network.

Note: Any arbitrary central node (Ego) which connects other nodes (Alters) in the network (solid lines) and connectivity among alters (dashed lines). Connectivity with rest of the network is represented with dashed-dotted lines.

Definition 3.1. (Level 1.0 ego network). If $G(V,E)$ is a network graph, then the ego network η or $\eta^{1.0}$ is defined as sub-graph $g(u,V_u,E_u)$ with respect to ego node u such that if $\exists v \in V$ then $v \in V_u$ iff $\exists (u,v) \in E$ and $\forall (x,y) \in E_u$ satisfies following conditions:

1. $x = u$
2. $y \in V_u$

Definition 3.2. (Level 1.5 ego network). If $G(V,E)$ is a network graph, then the ego network $\eta^{1.5}$ is defined as sub-graph $g(u,V_u,E_u)$ with respect to ego node u such that if $\exists v \in V$ then $v \in V_u$ iff $\exists (u,v) \in E$ and $\forall (x,y) \in E_u$ satisfies following conditions:

1. $x = u \vee x \in V_u$
2. $y \in V_u$

Definition 3.3. (Level 2.0 ego network). If $G(V,E)$ is a network graph, then the ego network $\eta^{2.0}$ is defined as sub-graph $g(u,V_u,E_u)$ with respect to ego node u such that if $\exists v \in V$ then $v \in V_u$ iff $\exists (u,v) \in E$ and satisfies following conditions:

1. $\forall(x,y) \in E_u$ implies $(x = u) \vee (x \in V_u)$ and $y \in V_u$

2. $V'_u = \bigcup_{\forall y \in V_u}(V_u, V_y)$ and $E'_u = \bigcup_{\forall y \in V_u}(E_u, E_y)$, where $g(V_y, E_y)$ is $\eta^{1.0}$

3. $V_u = V'_u$ and $E_u = E'_u$

3.1.2 Mutual Interest and Relationship

An ego network comprises several relationships involving ego and different alters. Considering the personal preference or personal interest factor, such relationship is viewed as two-way engagement of personal interest from both ego and alter's side as shown in the Figure 3.2. The natural social community has to be started evolving somewhere within the network. Suppose, a community has started to evolve from any arbitrary person (ego). Definitely, the next probable candidates for the community are the $\eta^{1.0}$ alters, which has only one member at present i.e. the ego. From the set of $\eta^{1.0}$ alters of ego, only those alters will qualify for the entrance to the community which involve stronger relationship with the ego. Strong relationship implies intensive mutual interest from both ego and alter's side. Without the interest of alter, the relationship between ego and alter cannot be stronger. This indicates the backward interest of alters are more decisive for their inclusion into the community. Backward interest of alters with the ego can only be confirmed, if collective forward interest of their own $\eta^{1.0}$ alters (excluding current ego) is less.

The above case describes the situation when a community has only one member i.e. the ego. Suppose, the community has more than one member. In this situation, any $\eta^{1.0}$ alter r of an ego (member of the community) can qualify as a community member, if the collective strength of relationships with all the members of the community to r is high. Again, the collective strength of relationships with r can be confirmed if the collective backward interest of r to all members of the community is more than the collective forward interest of r with the rest of the $\eta^{1.0}$ alters of r. Hence, it is the $\eta^{1.0}$ alter r of an ego (member

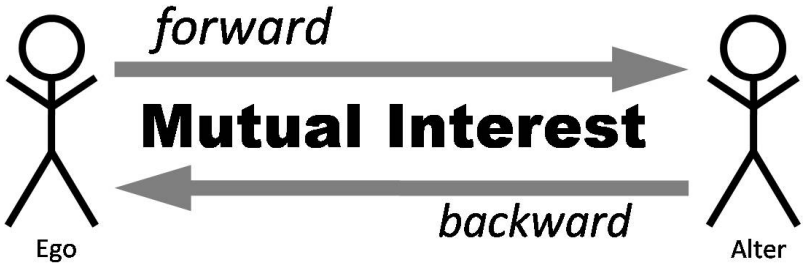

Figure 3.2: Mutual interest in the relationship between two persons.

Note: Personal interest of ego in the relationship with the alter and personal interest of alter in the relationship with the ego with respect to ego are interpreted as *forward* and *backward* personal interest respectively.

of the community), who is crucial for the decision whether it should be included into any community rather than deciding only by the members of the community. Considering this fact, a node level property called *Reachability* is defined for inclusion of any node r into any community C_i as follows:

$$R(r,C_i) = \frac{\sum\limits_{x \in (N_r \cap C_i)} \delta(r,x)}{\sum\limits_{x \in N_r} \delta(r,x)} \tag{3.1}$$

where, N_r is the list of $\eta^{1.0}$ alters of r or simply neighbors of r, $\delta(r,x)$ represents the strength of connection between alter x and r. Reachability of r tells the ability of r to reach out the members of the community with respect to its total known persons.

Community members have to be distinguished from rest of the network. Community members are distinguished mostly based on the connectivity pattern of the network. Naturally, persons having more connectivity among them are more likely to remain in the same community. More the connectivity among persons in the group and lesser the connectivity with rest of the network, the community becomes more distinguishable and isolated from the rest of the network. In this context, a community level property called *Isolability* is defined to measure the ability of any community to isolate itself from rest of the network

by considering the strength of connection or relationship. Isolability of any community C_i is defined as follows:

$$I(C_i) = \frac{\sum_{x,y \in C_i} \delta(x,y)}{\sum_{x,y \in C_i} \delta(x,y) + \sum_{x \in C_i, y \notin C_i} \delta(x,y)} \tag{3.2}$$

where, x and y are nodes, $\delta(x,y)$ represents the strength of connection between x and y. With these node level property and community level property, community structure is defined as below.

Definition 3.4. (Community). Given a network graph $G(V,E)$ with V nodes and E connections, a sub graph $g(V_c,E_c)$ is called a community C_i if satisfies following properties:

Property 1: (Node Level). $\forall v \in C_i$ have at least α amount of Reachability to C_i.

Property 2: (Community Level). C_i has at least β amount of Isolability with G.

Here, α and β are two parameters to specify minimum Reachability of each node in the community and the minimum Isolability of the community respectively.

3.2 Proposed Approach

In this section, proposed algorithm for detecting community structure in networks is presented. The algorithm has two phases Expansion and Dissolution. In expansion phase, identify possible communities and expand those accumulating new members. In dissolution phase, communities identified in the expansion phase are re-examined and unstable pseudo communities are dissolved into other stable communities. Detail about both these phased are explained below.

Algorithm 3.1: Expansion(G, α)

Input: $G(V,E), \alpha$
Output: C // Community list.
$P \leftarrow V$
$i \leftarrow 1$ // Community id.
$C_i \leftarrow \{empty\}$
while $\exists v \in P$ **do**
  $r \leftarrow$Select v from P with maximum connections and remove it from P
  $C_i \leftarrow \{C_i \cup r\}$
  while $r \neq -1$ **do**
    $N_r \leftarrow \eta^{1.0}$ alters of r that are $\notin C_i$
    $comExtend \leftarrow \{C_i \cup N_r\}$
    $R \leftarrow \forall v \in N_r$ compute Reachability with respect to *comExtend* using the Equation 3.1
    $flag \leftarrow 0$ //Indicator to check addition of any $v \in N_r$ to C_i.
    for all $v \in N_r$ **do**
      if $R_v \geq \alpha$ **then**
        //R_v is Reachability of node v.
        if $v \in P$ **then**
          // If v not assigned to any community.
          $C_i \leftarrow \{C_i \cup v\}$
          Remove v from P
          $flag \leftarrow 1$
        else
          if $v \notin C_i$ **then**
            // If v was already assigned to other community.
            Use tie breaking rule
            $flag \leftarrow 1$ if v added to C_i
          end if
        end if
      end if
    end for
    if $\nexists R_v \geq \alpha$ **and** $flag = 0$ **then**
      //Expansion of current community ends.
      $r \leftarrow -1$
    else
      $r \leftarrow$Select v from N_r with minimum R_v which are added to current community
    end if
  end while
  Add C_i to C
  $i \leftarrow i+1$ //Next community id
  $C_i \leftarrow \{empty\}$ //Start new community
end while
return C

3.2.1 Expansion Phase

This phase mainly does two tasks. The first task is to identify new community and the second task is to expand the identified community. The node level property *Reachability* is used to expand communities. Algorithm 3.1 outlines the process of identification of new communities and their expansion. The process starts with new empty community. A node r is selected from the list of unassigned nodes P, which has highest connections and include it into the new community. The community is expanded through $\eta^{1.0}$ alters N_r of r which are expected to include in the closed group of r. Hence, a temporary extended community is prepared by adding nodes in N_r to the new community (line 11). Reachability of all nodes in N_r with respect to this extended community is computed (line 12). Nodes having at least α amount of Reachability to the extended community are added to the new community if those nodes are previously not assigned to other community and remove from P. If any node v has been already assigned to community C_j and has to decide whether v has to be remain in C_j or it has to be reassigned to another community C_i. Such ties between communities C_j and C_i are broken as follows. First compute the change in Isolability of C_j and C_i (using Equation 3.2) with respect to presence and absence of v in the respective communities as follows:

$$d_j = I(C_j \cup v) - I(C_j \backslash v) \tag{3.3}$$

$$d_i = I(C_i \cup v) - I(C_i \backslash v). \tag{3.4}$$

Here, $I(C_j \cup v)$ and $I(C_i \cup v)$ are Isolability of C_j and C_i respectively in presence of v, $I(C_j \backslash v)$ and $I(C_i \backslash v)$ are Isolability of C_j and C_i respectively in absence of v. Then decide whether the node v will join community C_i or community C_j based on following rules:

Rule 1: If $d_j \geq d_i$, v will remain in C_j.

Rule 2: If $d_j < d_i$, v will reassign to C_i.

A node is selected from the list N_r, which are recently added to the current community and have least Reachability (line 34). The current community is further expanded through $\eta^{1.0}$ alters of this newly selected node. Expansion of current community ends when there is no node left in the N_r which has at least α amount of Reachability or no node has been added from N_r to the current community (line 32). Starts new community if any node left in the unassigned node list P.

3.2.2 Dissolution Phase

Communities identified in the expansion phase are examined again in this phase. Communities which are not able to isolate themselves from rest of the network are treated as unstable and pseudo communities. If any community has Isolability less than β, then the community is considered as unstable. Any unstable community C_r is dissolved into other suitable and stable community. Suitable stable community with respect to unstable community C_r is identified as follows. Algorithm 3.2 outlines pseudocode for dissolution of unstable communities. First of all examine the appearance of $\eta^{1.0}$ external alters with respect to all nodes in C_r (lines 6-9). Identify the community containing $\eta^{1.0}$ external alter that appeared highest number of times with respect to all nodes in C_r (line 10). If more than one communities contain such external alters, the community that benefits more by dissolving unstable community C_r into it is considered as suitable community for dissolution (lines 12-19). After dissolution of an unstable community into stable community it is removed from the community list (line 20).

Algorithm 3.2: Dissolution(C, β)

Input: C, β
Output: C' // Dissolved community list.
for all $C_r \in C$ **do**
  $I_r \leftarrow$ Compute Isolability of community C_r using Equation 7.6
  if $I_r < \beta$ **then**
    $X \leftarrow$ Find set of all external $\eta^{1.0}$ alters of all nodes $v \in C_i$
    $Xcount \leftarrow$ Count number of times $\forall u \in X$ appears in $\eta^{1.0}$ alters with respect to all $v \in C_i$
    $maxc \leftarrow$ Maximum count obtained in $Xcount$
    $Mx \leftarrow$ Set of nodes in X which has obtained $maxc$ value
    $Mc \leftarrow$ Set of communities which includes any node of Mx
    $Mdif \leftarrow -\infty$
    for all $C_k \in Mc$ **do**
      $dif \leftarrow I(C_k \cup C_r) - I(C_k)$
      if $Mdif > dif$ **then**
        $Mdif \leftarrow dif$
        $d \leftarrow k$
      end if
    end for
    $C_d \leftarrow (C_d \cup C_r)$
    $C' \leftarrow (C \backslash C_r)$
  end if
end for
return C'

3.3 Empirical Analysis

3.3.1 Experimental Setup

Performance of ENBC is compared with that of six state-of-the-art community detection algorithms FastU [17], HC-PIN [203], LeadF [173], LICOD [97], RandW [183] and SCAN [216]. Among these six algorithms HC-PIN, LICOD and SCAN have specific parameters to control community structure. Values of these parameters are considered as indicated in the respective literature. HC-PIN has two parameters λ and os. The parameter λ determines community size and os determines overlapping of communities. Parameters $\lambda = 1$ and $os = 0.0$ for non-overlapping communities. LICOD has three parameters

σ for determining leaders, δ for merging two communities and ε for overlapping. Values of these parameter are considered as follows: $\sigma = 0.7$, $\delta = 0.28$ and $\varepsilon = 0.0$ for non-overlapping communities. SCAN has two parameters ε for similarity limit and μ for number of neighbors to be processed, which are considered as follows: $\varepsilon = 0.7$ and $\mu = 2$ for processing two neighbors. Proposed algorithm ENBC also have two parameters α and β. After rigorous analysis of both the parameters (see subsection 3.3.5), α value 0.5 and β value 0.45 are considered for detecting communities.

Performance of ENBC is evaluated both in terms of accuracy metrics and quality metrics. Five accuracy metrics are considered for measuring accuracy of communities, which include popularly used NMI [184], ARI [157], F-measure, Purity [126] and Entropy [225]. Four quality metrics are considered for evaluating quality of communities, which include widely used Modularity [143], Coverage [21], ExtD [169] and AVI (see subsection 7.2.3 for detail). Besides these metrics, number of communities (NoC) is also presented. Seven real-world networks are considered (see section 2.3 for detail explanation). Among those, four networks Dolphin [122], Football [61], Karate [218] and Strike [132] are with known ground truth communities. Remaining three networks GR-QC [112], HEP-TH [112] and Wiki-Voter [111] do not have ground truth, those are used only for measuring quality of communities. Two LFR graphs (synthetic networks) [109], LFR 1 and LFR 2 with 128 nodes and 1000 nodes respectively are also considered for experiments.

3.3.2 Analyzing Accuracy

Results obtained in terms of accuracy metrics on all the datasets, where ground truths are available presented in Table 3.1. Clearly, on Football network ENBC shows higher NMI, ARI and Purity than all other competitors. SCAN also shows higher NMI, ARI and Purity than other algorithms, but lagging behind ENBC. However, HC-PIN shows highest

Table 3.1: Accuracy metric values in various datasets having ground truth communities.

Datasets	Algorithms	NMI	ARI	Purity	F-measure	Entropy	NoC
Football (12)	LICOD	0.5088	0.0924	0.1217	0.0655	0.2846	58
	FastU	0.8787	0.7913	0.8609	0.4554	0.1507	10
	SCAN	0.9079	0.8393	0.913	0.3672	0.0957	12
	HC-PIN	0.898	0.8302	0.8957	0.6875	0.1172	11
	RandW	0.7292	0.5151	0.6435	0.3045	0.2808	10
	LeadF	0.6411	0.4306	0.6	0.2544	0.4001	9
	ENBC	0.9454	0.8716	0.9391	0.6608	0.1626	10
Strike (3)	LICOD	0.468	0.3712	0.2502	0.0385	0.3043	6
	FastU	0.7704	0.6647	0.9583	0.6813	0.1233	4
	SCAN	0.766	0.7486	0.8583	0.7666	0.1233	3
	HC-PIN	0.766	0.7486	0.8583	0.7666	0.1233	3
	RandW	0.751	0.6333	0.5	0.2	0.0724	5
	LeadF	0.5653	0.2541	0.9167	0.0905	0.1706	7
	ENBC	0.8841	0.7978	0.875	0.75	0	4
Dolphin (2)	LICOD	0.3517	0.1232	0.1935	0.1534	0.5031	11
	FastU	0.4838	0.3464	0.5806	0.2336	0.1676	5
	SCAN	0.5725	0.4579	0.4194	0.0958	0.3902	3
	HC-PIN	0.8888	0.9348	0.9839	0.9818	0.11	2
	RandW	0.4157	0.1846	0.2419	0.2222	0.1112	9
	LeadF	0.3226	0.068	0.2097	0.0361	0.1613	17
	ENBC	0.7803	0.8721	0.9677	0.9659	0.2029	2
Karate (2)	LICOD	0.1604	0.0514	0.5588	0.0801	0.32	6
	FastU	0.5866	0.4619	0.9706	0.1964	0.1461	4
	SCAN	0.7483	0.5994	0.7647	0.12	0.1623	4
	HC-PIN	0.6194	0.5644	0.9706	0.2167	0.1529	4
	RandW	0.8365	0.8823	0.9706	0.9704	0.1614	2
	LeadF	0.4522	0.3362	0.9412	0.131	0.1765	8
	ENBC	0.8372	0.8823	0.9706	0.9706	0.1614	2
LFR 1(28)	LICOD	0.7393	0.3471	0.388	0.0504	0.3995	9
	FastU	0.9613	0.9442	0.972	0.003	0.038	28
	SCAN	-	-	-	-	-	-
	HC-PIN	0.473	0.0916	0.162	0.0186	0.6814	4
	RandW	0.8334	0.5019	0.544	0.024	0.268	15
	LeadF	0.9613	0.9442	0.972	0.003	0.038	28
	ENBC	0.9613	0.9442	0.972	0.003	0.038	28
LFR 2(4)	LICOD	0.5772	0.328	0.5	0.125	0.3002	3
	FastU	0.9498	0.9583	0.9844	0.9846	0.0503	4
	SCAN	-	-	-	-	-	-
	HC-PIN	0.5772	0.328	0.5	0.3333	0.5944	2
	RandW	0.5772	0.328	0.5	0.3333	0.5944	2
	LeadF	0.9498	0.9583	0.9844	0.9846	0.0503	4
	ENBC	0.9498	0.9583	0.9844	0.9846	0.0503	4

F-measure value. ENBC is slightly behind HC-PIN. Though, ENBC shows lower entropy value, SCAN and HC-PIN show little bit lower values. Performance of LICOD, RandW and LeadF is not good with respect to all metrics. In fact, LICOD produces quite larger NoC than the ground truth and almost all metrics have worst values. On the contrary, other algorithms including ENBC produce similar NoC. On Strike network also ENBC outperforms almost in all metrics. Performances of FastU, SCAN and HC-PIN are quite good but not better than ENBC. LICOD, RandW and LeadF show worst performance. Though, LeadF and FastU acquire higher Purity value their overall performance is not better. Interestingly, ENBC has acquired Entropy value zero, which is plus point for correctness of predicted communities. HC-PIN shows better performance than all algorithms on Dolphin network. ENBC also shows much better performance than other algorithm except HC-PIN. LICOD and LeadF produce large NoC and also show worst performance. On Karate network, ENBC outperforms all of the six algorithms. RandW shows almost similar results as ENBC, whereas other algorithms specially LICOD and LeadF show poor performance. In both synthetic networks, LFR 1 and LFR 2, algorithms HC-PIN, LICOD and RandW show poor performance. For both networks, these algorithm produces lesser NoC than actual number of communities. However, FastU, LeadF and ENBC show similar performance in both networks. Overall, ENBC performs better than all other algorithms in terms of most accuracy metrics. This implies that communities detected by ENBC are highly accurate compared to other algorithms.

3.3.3 Analyzing Quality

Results obtained in terms of quality metrics on the datasets, where ground truth is available are presented in Table 3.2. On Football network, all algorithms show similar NoC except LICOD. Clearly, metric values in LICOD are poorer in comparison to other algorithms. On Dolphin network also LICOD and LeadF produce larger NoC and their

Table 3.2: Quality metric values for datasets having ground truth communities.

Datasets	Met.	Algorithms						
		LICOD	FastU	SCAN	HC-PIN	RandW	LeadF	ENBC
Football (12)	Cov	0.0881	0.708	0.6754	0.6998	0.6215	0.5514	0.708
	ExtD	0.0891	0.0302	0.0333	0.0308	0.0401	0.0476	0.0304
	Q	0.0798	0.6413	0.611	0.6338	0.5629	0.4993	0.6413
	AVI	0.0227	0.5469	0.4622	0.5268	0.4	0.342	0.5429
	NoC	58	10	12	11	10	9	10
Strike (3)	Cov	0.6316	0.8684	0.9211	0.9211	0.7895	0.6053	0.8684
	ExtD	0.0769	0.0249	0.0167	0.0167	0.0369	0.061	0.0246
	Q	0.5391	0.7278	0.7666	0.7666	0.6589	0.517	0.7258
	AVI	0.2589	0.7457	0.8552	0.8552	0.5848	0.458	0.7487
	NoC	6	4	3	3	5	7	4
Dolphin (2)	Cov	0.3396	0.7547	0.5975	0.9623	0.4906	0.4214	0.956
	ExtD	0.0643	0.0264	0.042	0.0071	0.0488	0.0523	0.0081
	Q	0.3015	0.648	0.5062	0.8216	0.4313	0.3777	0.8161
	AVI	0.1362	0.5601	0.4979	0.9129	0.2825	0.2196	0.902
	NoC	11	5	3	2	9	17	2
Karate (2)	Cov	0.8462	0.7308	0.5769	0.7564	0.8718	0.4615	0.8718
	ExtD	0.0738	0.0506	0.0686	0.0496	0.0351	0.0911	0.0347
	Q	0.6053	0.5388	0.401	0.5416	0.6341	0.354	0.6226
	AVI	0.1865	0.5579	0.4785	0.5272	0.7684	0.2314	0.7726
	NoC	6	4	4	4	2	8	2
LFR 1(28)	Cov	0.988	0.972	-	0.996	0.985	0.972	0.972
	ExtD	2.83E-05	5.82E-05	-	1.43E-05	3.33E-05	5.82E-05	5.82E-05
	Q	0.9589	0.953	-	0.9602	0.9559	0.953	0.953
	AVI	0.9664	0.9437	-	0.988	0.959	0.9437	0.9437
	NoC	9	28	-	4	15	28	28
LFR 2(4)	Cov	0.9843	0.9843	-	1	1	0.9843	0.9843
	ExtD	3.91E-04	3.26E-04	-	0	0	3.26E-04	3.26E-04
	Q	0.8275	0.8275	-	0.7768	0.7768	0.8275	0.8275
	AVI	0.9695	0.9693	-	1	1	0.9693	0.9693
	NoC	3	4	-	2	2	4	4

corresponding metrics show degraded values. Opposite to this fact, smaller NoC resulting better quality metrics are also notable. On Strike network SCAN and HC-PIN, on Dolphin network HC-PIN and ENBC, and on Karate network RandW and ENBC clearly have the best quality metric values. Both of these two facts are also seems to be true for synthetic networks as well. For LFR 1 original network had 28 communities. LICOD, HC-PIN and RandW produce comparatively lower NoC. Accordingly, one can notice the better quality metric values. Note that the best quality metric values obtained for HC-PIN are the cases when NoC is small. On the contrary, larger NoC as produced by FastU, LeadF and ENBC show good quality metric values those are poorer than other three, while NoC is exactly same as original. Similar observation can also be made for LFR 2 data set. However, on Strike, Dolphin and Karate networks, it is clear that the exact NoC as in original network show best quality metric values. This is because in original network these data sets had very less (2 or 3) NoC. These facts are noticeable because the ground truth is known. As most of the real-world networks do not have ground truth so it is difficult for deciding which optimal values of quality metric can be considered as better. One possible way to overcome this issue if approximate NoC by observing results of all algorithms, then one can easily decide on which optimal quality metric values are better.

Exploring further the above notion, examined whether approximation of actual NoC by observing NoC produced by multiple algorithms. The results presented in Table 3.2 are analyzed in this context by assuming that the exact NoC of these networks are unknwon. For Football network FastU, RandW and ENBC produce 10 communities. SCAN, HC-PIN and LeadF also produce almost same NoC. However, LICOD produces 58 communities, which is different from other algorithms. Thus, one can have intuitive sense that the Football network may have nearly 10 communities. Now, with this primary approximated NoC it can clearly explained that FastU and ENBC show better quality metric values indicating better community structure. Similarly, for Strike, Dolphin, Karate, LFR 1 and

Table 3.3: Quality metric values for datasets whose ground truth are unavailable.

Datasets	Metrics	Algorithms						
		LICOD	FastU	SCAN	HC-PIN	RandW	LeadF	ENBC
GR-QC	Cov	0.7654	0.8967	0.7538	0.8242	0.5532	0.5769	0.8341
	ExtD	1.8E-04	1.1E-04	2.61E-04	1.9E-04	4.72E-04	4.5E-04	1.8E-04
	Q	0.6503	0.8814	0.7392	0.8093	0.548	0.5698	0.8191
	AVI	0.7509	0.9821	0.6951	0.8119	0.6553	0.3865	0.8098
	NoC	615	390	844	698	862	1123	712
HEP-TH	Cov	0.6918	0.7993	0.567	0.7191	0.4816	0.4303	0.7946
	ExtD	1.3E-04	1.1E-04	2.3E-04	1.5E-04	2.8E-04	3.1E-04	1.2E-04
	Q	0.6889	0.7967	0.5651	0.7169	0.4808	0.4294	0.7917
	AVI	0.6584	0.9719	0.5699	0.7709	0.5356	0.2991	0.7573
	NoC	570	474	1559	986	1504	2445	1055
Wiki-Vote	Cov	1	-	0.5792	1	0.3058	0.0677	0.9999
	ExtD	1.1E-05	-	0.0017	0	0.0035	0.0037	2.6E-05
	Q	0.9047	-	0.5192	0.9047	0.2712	0.0666	0.9046
	AVI	0.9771	-	0.384	1	0.0993	0.0536	0.8727
	NoC	25	-	95	24	377	1382	33

LFR 2, NoC can be approximated as 3, 2, 2, 28 and 4 respectively. Hence, without knowing original NoC, it can be easily asserted by considering approximated NoC that LICOD and LeadF produce worst quality communities in almost all networks.

Now, similar approximation is done to analyze quality of communities produced in networks whose original community structure actually unknown. Results on three such kind of networks are presented in Table 3.3. For GR-QC network, most of the algorithms produce NoC higher than 600 and lower than 900. Hence, it is quite reasonable to consider approximated NoC between 600 and 900. Considering this range, clearly ENBC produces best quality communities in GR-QC network as indicated by quality metric values. HC-PIN also produces almost same result as ENBC. FastU produces optimal values of all quality metrics. However, FastU produced least NoC, which is reasonably lesser than most of the algorithms. As explained above, high quality metric value does not imply

good quality communities unless NoC is reasonable. This fact is also observed by Brandes et al. [21] and they also suggested reasonable NoC as constraint to quality metrics. Kanawati [97] pointed that quality metric does not correspond to best partitioning specially as done by accuracy metrics. Steinhaeuser et al. [183] stated that quality metric such as Modularity does not strictly imply good or bad partitioning. However, as optimal value of quality metric is supposed to indicate better community structure by definition. Therefore, optimal value with reasonable NoC can be treated as better partitioning. Now, with constrain to NoC, it is noticeable that quality of community structure is not good even though FastU produces optimal value.

Similarly, for HEP-TH network also FastU produce least NoC with optimal quality metric values, which indicate poor communities. In this case again ENBC seems to be producing best communities and NoC producing is also quite reasonable. For Wiki-Vote network, most of the algorithms produced small NoC. However, RandW and LeadF produced very large NoC in comparison to other algorithms. The Wiki-Vote network has 24 disconnected components, one of them is huge dense component and rest of 23 are smaller. Therefore, even though HC-PIN produces 24 communities, it actually produces individual components and the unpartitioned huge component. That means each component entirely representing a community. Hence, obtained optimal values for all the metrics in HC-PIN. LICOD also produces 25 communities, i.e. huge component is partitioned into only two communities, which also cannot be considered logically as better partitioning. SCAN also indicates the same. On the contrary, ENBC produces 33 communities, i.e. large component is partitioned into 9 communities, which is quite reasonable for such highly connected component. Moreover, ENBC shows better quality metric values indicating high quality communities. LeadF produces worst communities in all three data sets with highest NoC. Overall, considering the constraint to NoC resulted, ENBC seems to be producing high quality communities in all three networks.

Table 3.4: MCDM ranking score obtained with 75% accuracy and 25% quality.

Algorithms	Datasets					
	Football	Strike	Dolphin	Karate	LFR 1	LFR 2
LICOD	0.4000	0.2550	0.2833	0.1865	0.4219	0.2232
FastU	0.5568	0.6281	0.5296	0.4932	**0.8652**	**0.8147**
SCAN	0.5690	0.6084	0.4742	0.5440	-	-
HC-PIN	0.5760	0.6084	**0.7967**	0.5222	0.1300	0.2675
RandW	0.5059	0.4206	0.5474	0.6044	0.6010	0.2675
LeadF	0.4669	0.5227	0.3247	0.5126	**0.8652**	**0.8147**
ENBC	**0.5789**	**0.6420**	0.7684	**0.6120**	**0.8652**	**0.8147**

Note: Higher score indicates more inclination of algorithm towards accuracy. Algorithms acquiring higher rank are indicted in bold.

3.3.4 MCDM Ranking

MCDM ranking is done for accumulating all of accuracy metrics and quality metrics under one single score. TOPSIS method explained by Kou et al. [103] is considered for the ranking. The TOPSIS method has the privilege to assign different weights to each criterion (i.e. metrics). Since the objective is to assure accuracy in communities so 75% of weights are assigned to accuracy and rest of the 25% weights are assigned to quality metrics. Weights assigned to both accuracy and quality are equally distributed over respective metrics. Equally distributed in the sense that there are five accuracy metrics and 75% of total weight 1 is 0.75, which is equally distributed over all five accuracy metrics. Hence, each metrics will get weight $\frac{0.75}{5} = 0.15$ to contribute in ranking. Similarly, four quality metrics will get weight $\frac{0.25}{4} = 0.0625$ each to contribute in ranking. To analyze effectiveness of communities generated by different algorithms including the proposed ENBC with respect to accuracy, variation in the ranking score with increment in percentage of accuracy is analyzed. This analysis is formally referred as RITA analysis (see section 7.3 for more detail). The accuracy metric weights are varied from 25% to 75%, and apparently quality metric weights varied from 75% to 25%.

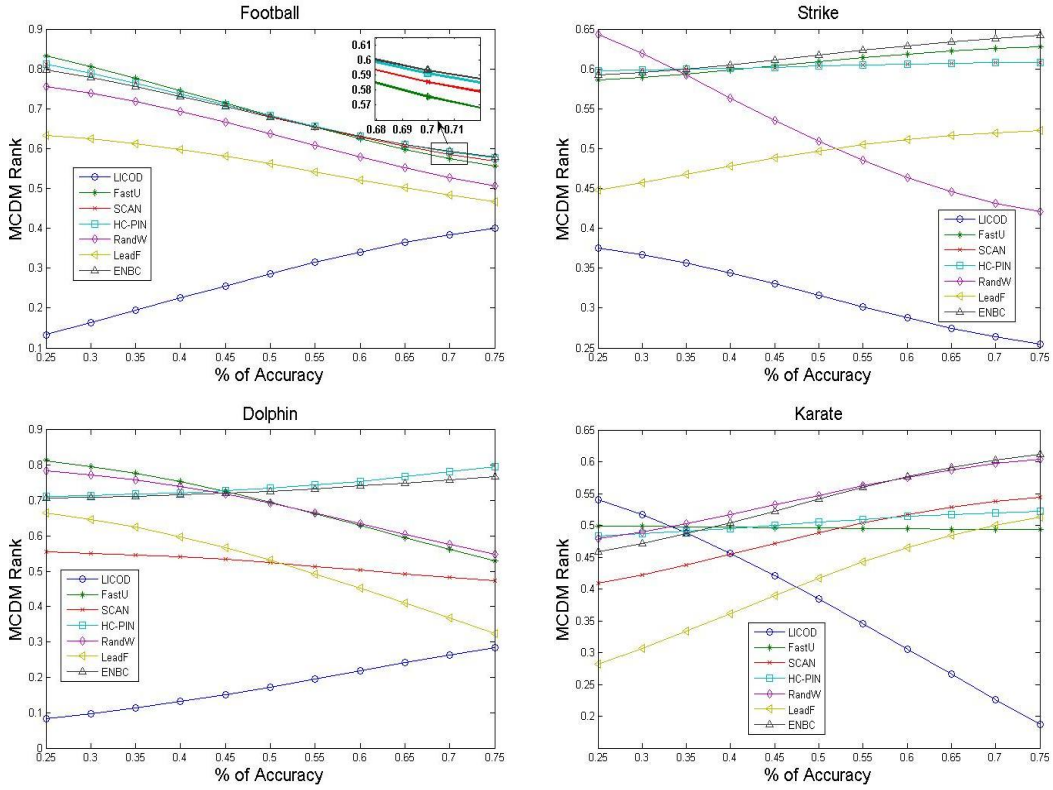

Figure 3.3: MCDM ranking acquired by each algorithm in real world known network with variation of accuracy contribution. Higher scores justify tendency of algorithm's inclination towards accuracy.

Table 4.10 presents MCDM ranking obtained for the communities predicted by different algorithms. Since, more weights are allocated to accuracy metrics so scores obtained will indicate algorithm's inclination towards accuracy. Clearly, ENBC shows higher scores for all networks except Dolphin network, which indicates ENBC produces highly inclined communities towards accuracy. For Dolphin network, HC-PIN acquires highest score because both accuracy and quality metrics were higher, which was also noticed in the earlier analysis. ENBC acquires slightly lower score than the HC-PIN, but score is much more significant than rest of the algorithms. LICOD and LeadF show very poor inclination towards accuracy in most of the data sets and it is obvious since their accuracy as well as quality metric values were poor.

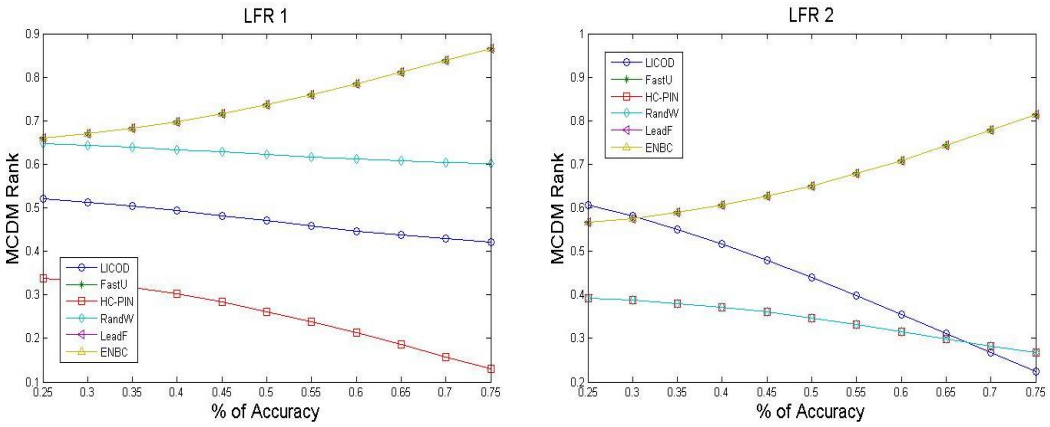

Figure 3.4: MCDM ranking acquired by each algorithm in synthetic network with variation of accuracy contribution.

MCDM ranking with variation of percentage of accuracy contribution in the accumulated score is presented in Figure 7.9. FastU, SCAN, HC-PIN and ENBC show almost similar characteristics for all the networks. Initially, when accuracy contribution given weightage 25%, FastU and HC-PIN shows higher scores than ENBC, which is obvious as both these algorithms produced communities with low accuracy and comparatively higher quality than ENBC. When accuracy given 25% weightage in ranking, apparently quality contribution becomes 75% so quality metrics will have more impact on the score. As gradually percentage of accuracy increases ENBC acquires higher scores. For Dolphin network, as both quality and accuracy of HC-PIN was higher so ENBC cannot overtake even when accuracy contribution increases up to 75%. However, ENBC constantly follows HC-PIN till increment of accuracy contribution up to 75%. LICOD, LeadF and RandW are no where in the competition with ENBC. Though, initially some time they acquired higher scores than ENBC when accuracy contribution was low, most of the cases they shows significantly lower scores than ENBC. For synthetic network also reflects the same outcome as shown in Figure 3.4.

3.3.5 Parametric Analysis

Proposed ENBC algorithm has two parameters Alpha (α) and Beta (β). Dependence of algorithm's performance on α and β is analyzed. Main objective of this analysis is to determine suitable values of α and β for obtaining high quality and highly accurate communities. Too high or too low values of α and β produce insignificant communities so values ranged from 0.25 to 0.70 are considered for parametric analysis. Values of both α and β are increased with interval of 0.05 and paired them. This 0.05 increment yields 10 values for both α and β. Hence, resulted 10×10=100 pairs of α and βvalues. For each 100 pairs of values, communities obtained with ENBC algorithm is examined.

To evaluate results obtained with respect to different α and β pairs, four quality metrics (Modularity, Coverage, External Density and Average Isolability) and three accuracy metrics (NMI, ARI and Purity) are considered. In addition, NoC generated corresponding to each pair of α and β are also analyzed. Results obtained on Strike, Karate, Football and Dolphin networks for all 100 pairs of α and β values are presented in Figure 3.5, Figure 3.6, Figure 3.7 and Figure 3.8 respectively.

Observations: Clearly, high β value results high quality communities. This is because β is the threshold for assigning nodes of low quality communities to other communities which ensures good quality community structure after such assignment. But at the same time, higher values of β imply reduction of NoC. However, high α value results high NoC, provided β value is low. For high β values, the effect of α diminishes. Though high β value results some of the quality and accuracy measures better, but it causes deviation from actual NoC. Nevertheless, quality metrics may have better values depending on community structure. As far as accuracy is concern, it has dependency on NoC as indicated in the above analysis. If the NoC is different from actual NoC, it means communities

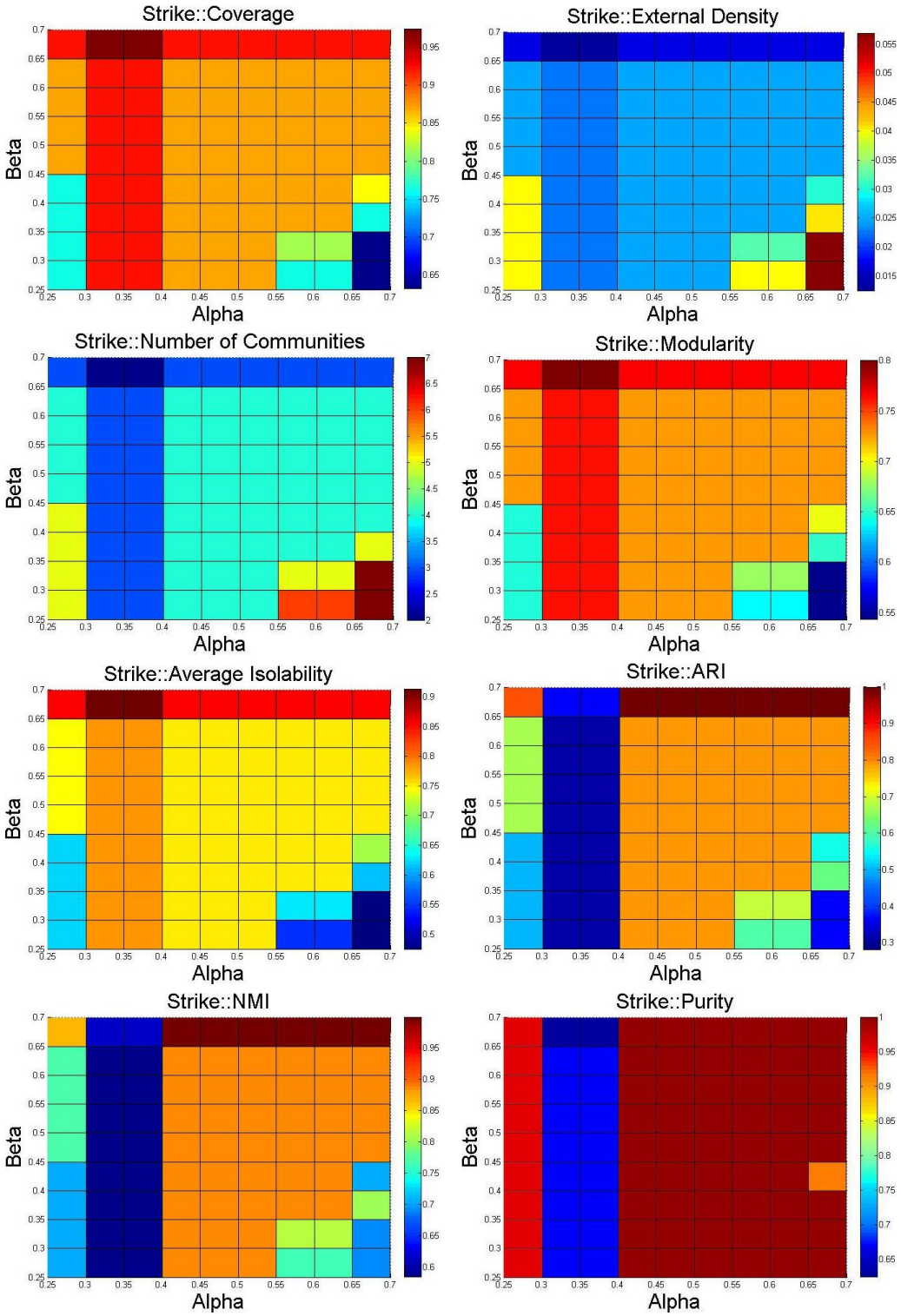

Figure 3.5: Safe zone predicted with different metrics corresponding to α and β ranges in Strike data set.

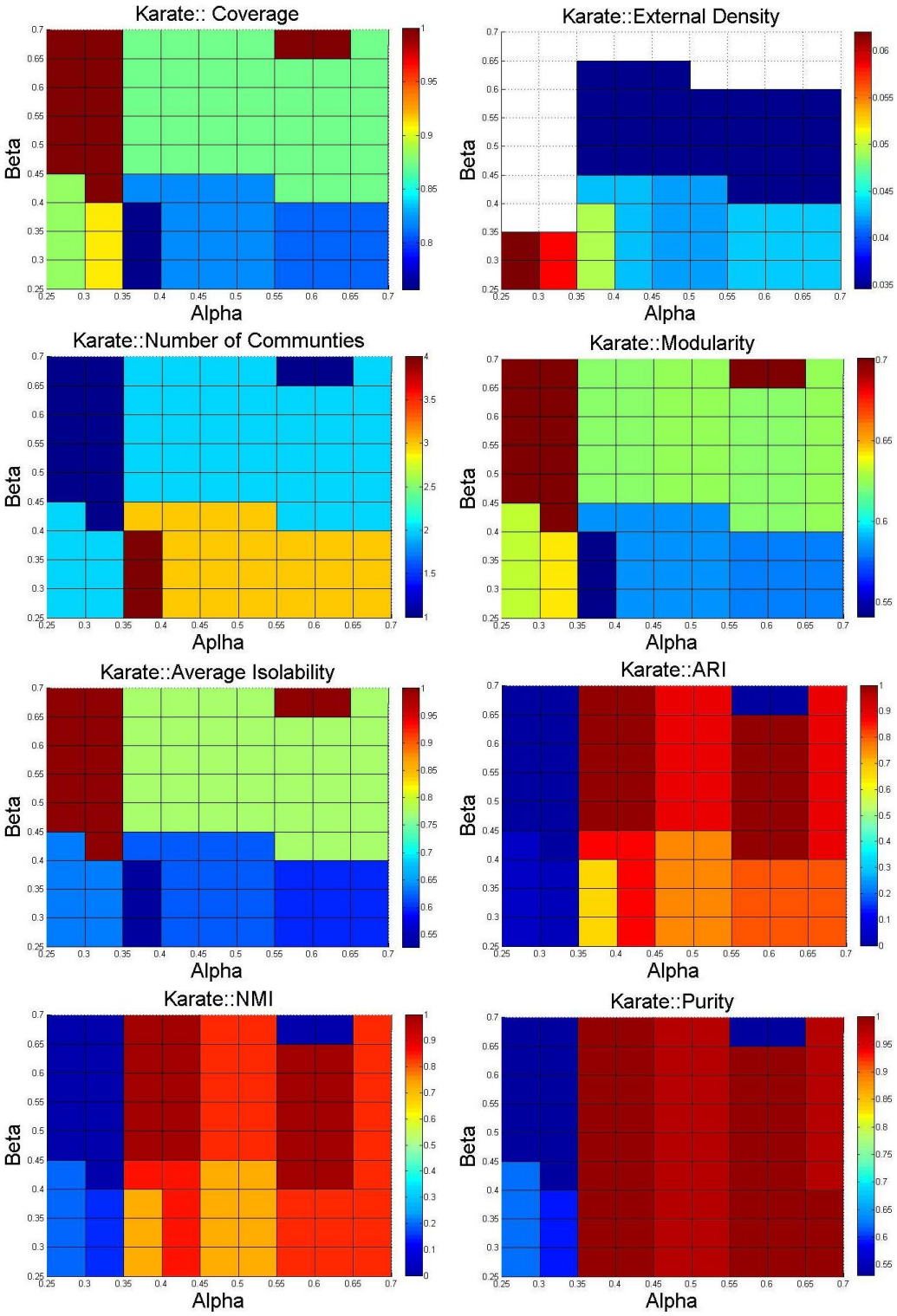

Figure 3.6: Safe zone predicted with different metrics corresponding to α and β ranges in Karate data set.

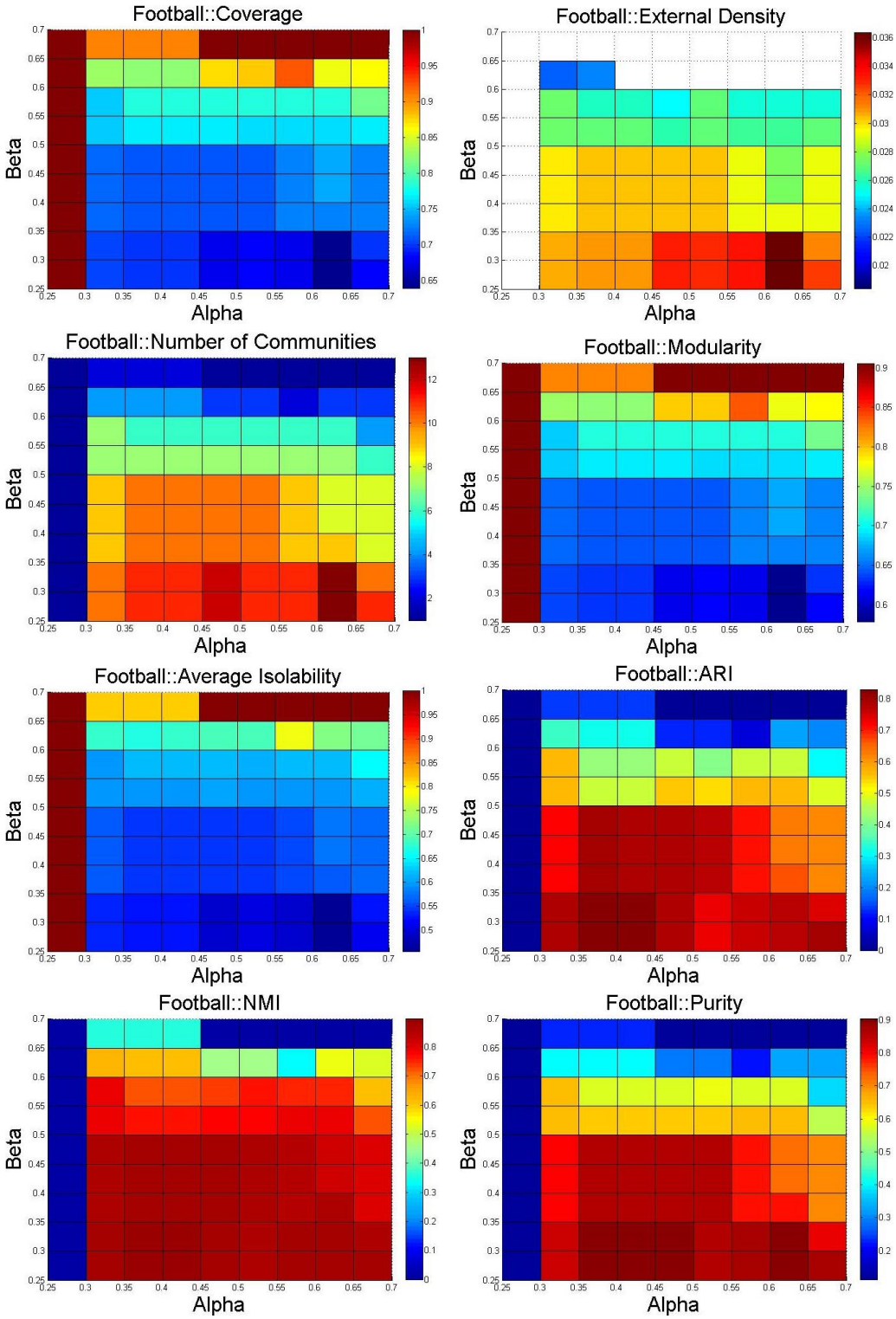

Figure 3.7: Safe zone predicted with different metrics corresponding to α and β ranges in Football data set.

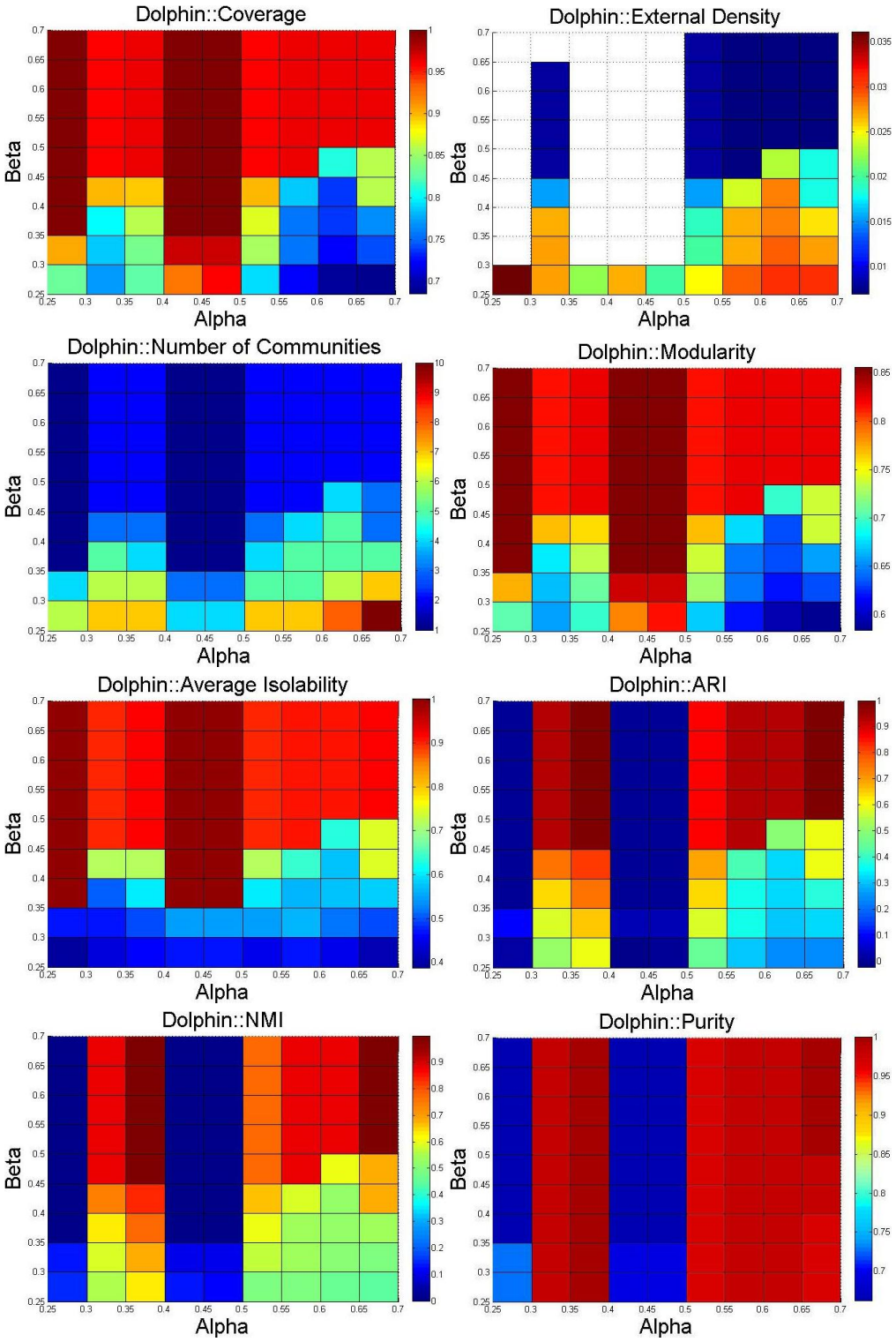

Figure 3.8: Safe zone predicted with different metrics corresponding to α and β ranges in Dolphin data set.

are identified wrongly. Very few nodes may be misplaced but identifying completely different community does not ensure accuracy. Even though various accuracy measures get high values, but it cannot justify the wrong prediction of communities. Also another point to be noted for all four networks, that for smaller NoC, quality metrics such as Modularity, Coverage and average Isolability are getting higher values (shown in Figure 3.5, Figure 3.6, Figure 3.7 and Figure 3.8). Moreover these metrics attain their highest possible value 1, which happens only when NoC also 1. Though, by definition highest value 1 ensures good quality communities but in real sense that is not the case, specially when already know the NoC more than 1. Considering all these aspects and focusing more on accuracy, safe zones for all the metrics with different networks are analyzed. The safe zones are overall ranges of α and β values, which ensure accuracy as well as quality of communities. It is clear from different safe zones that β values from range 0.45-0.55 and α values from range 0.3-0.5 are suitable for detecting finer communities.

3.4 Computational Complexity Comparison

The ENBC algorithm has two phases *Expansion* and *Dissolution*. In a given graph of n nodes and m edges, Expansion phase of ENBC processes k random nodes with higher degree for expanding communities (line 6). Note these k nodes are the initial members of the community that are processed after finishing expansion of predecessor community. Thus, k numbers of communities will generate in Expansion phase. Excluding these k nodes, Reachability of remaining $(n-k)$ are computed with cost $O(n)$ (line 12). Reachability computation requires maximum cost compared to rest of the part of Expansion phase. Therefore, cost of Expansion phase of ENBC is $O(n^2)$.

Table 3.5: Summary of complexity of community detection algorithms.

Algorithm	Complexity	Reference	Remarks
LICOD	$O(n^3)$	[97]	Smaller and very inaccurate communities
FastU	$\Omega(n^2)$	[17]	Communities obtained in several abstraction levels
SCAN	$O(m)$	[216]	Identifies hubs, outliers but inaccurate communities
HC-PIN	$O(d^2 \times m)$	[203]	Accurate communities, but for scale free network
RandW	$O(n^2 log\ n)$	[183]	Accurate communities in small network
LeadF	$O(n \times m)$	[173]	Smaller communities in dense network
ENBC	$O(n^2)$	This work	Highly accurate communities, but still bound by cost

Dissolution phase of ENBC computes Isolability of all k communities generated in Expansion phase and dissolves communities with lower Isolability than β to other communities. Cost of Isolability computation of any community C_i with n_i numbers of nodes is $O(n_i \times n)$. Thus, total cost incurred in Isolability computation of all communities is $O(\sum_{i=1}^{k} n_i \times n)$ i.e. $O(n^2)$. Isolability computation requires maximum cost compared to remaining the part of Dissolution phase. Therefore, cost of Dissolution phase of ENBC is also $O(n^2)$. Hence, overall computation cost of ENBC is $O(n^2)$.

Complexity of other existing algorithms are summarized in Table 4.1. Complexity of algorithms LICOD, FastU, RandW and LeadF are higher than ENBC. In terms of accuracy, communities identified with these algorithms are far behind ENBC. Complexity of SCAN is linear, but identified communities are very inaccurate. On the contrary, HC-PIN also draws linear time complexity for scale free network [203], and generates highly accurate communities. For scale free networks, the term average degree d in actual cost $O(d^2 \times m)$ becomes constant. Nevertheless, already proved that the ENBC produces more accurate communities than HC-PIN. Communities identified with ENBC are most accurate, which is the main strength of this algorithm. However, weak side of ENBC is its quadratic time complexity. Each algorithm has advantages and disadvantages. Therefore, user must be careful while selecting best suited algorithm as per the requirements. ENBC would be best option for the applications seeking accurate communities.

3.5 Conclusion

In this chapter, the factors that influence the accuracy of community detection algorithm are traced. Generally, dense connectivity among nodes is considered for defining communities, which is quite logical for ensuring quality of communities. As far as accuracy is concerned, dense connectivity cannot assure accuracy of communities because such definition does not give freedom to nodes for deciding their belongingness to any community. In this context, different properties of network that are commonly used in community detection algorithms are discussed in three levels of abstraction, node level, community level and network level. With substantive personalized view of ego network, defined a node level property *Reachability* and a community level property *Isolability*. Harnessing these two properties community structure is re-defined giving more freedom to nodes and proposed an ego network based algorithm called ENBC to detect communities. Main contributions are mentioned as follows.

- Performance of ENBC has been compared with six state-of-the-art community detection algorithms. Results obtained on real-world networks as well as synthetic networks are evident for the superiority of ENBC over other algorithms.
- Inclination of ENBC towards accuracy is comparatively higher than other algorithms as indicated in the analysis of MCDM ranking.
- Parametric analysis shows the effectiveness of ENBC in tackling diverse challenges such as network size, sparsely connected or densely connected networks with its flexible parameters α and β.
- Complexity of ENBC is $O(n^2)$, which is lower than several existing approaches.
- Major advantage of ENBC is that it identifies highly accurate communities.

Practical implication of community detection algorithms often faces trade-off between accuracy and quality of communities. The problem arises because most of the community detection algorithm utilizes only connectivity. This results in detection of high quality but very inaccurate communities. On the contrary, proposed ENBC algorithm deepens the role nodes along with connectivity in community detection process. Previous approaches give priority to only the connections or group of nodes for deciding membership of a node, whereas ENBC gives priority to node itself for such decision. The notion imitates natural way of community formation that gives privilege to identify communities accurately.