Local Graph Stability in Exponential Family Random Graph Models[*§]

Yue Yu[¶1], Gianmarc Grazioli[‖2], Nolan E. Phillips[**3], and Carter T. Butts[††‡‡1,4,5]

[1]Department of Computer Science, University of California, Irvine
[2]Department of Chemistry, San José State University
[3]Department of Sociology, Harvard University
[4]California Institute for Telecommunications and Information Technology
[5]Departments of Sociology, Statistics, and Electrical Engineering and Computer Science; and Institute for Mathematical Behavioral Sciences, University of California, Irvine

Abstract

Exponential family Random Graph Models (ERGMs) are widely used to model networks by parameterizing graph probability in terms of a set of user-selected sufficient statistics. Equivalently, ERGMs can be viewed as expressing a probability distribution on graphs arising from the action of competing social forces that make ties more or less likely, depending on the state of the rest of the graph. Such forces often lead to a complex pattern of dependence among edges, with non-trivial large-scale structures emerging from relatively simple local mechanisms. While this provides a powerful tool for probing macro-micro connections, much remains to be understood about how local forces shape global outcomes. One very simple question of this type is that of the conditions needed for social forces to stabilize a particular structure: that is, given a specific structure and a set of alternatives (e.g., arising from small perturbations), under what conditions will said structure remain more probable than the alternatives? We refer to this property as local stability and seek a general means of identifying the set of parameters under which a target graph is locally stable with respect to a set of alternatives. Here, we provide a complete characterization of the region of the parameter space inducing local stability, showing it to be the interior of a convex cone whose faces can be derived from the change-scores of the sufficient statistics vis-à-vis the alternative structures. As we show, local stability is a necessary but not sufficient condition for more general notions of stability, the latter of which can be explored more efficiently by using the "stable cone" within the parameter space as a starting point. In addition to facilitating the understanding of model behavior, we show how local stability can be used to determine whether a fitted model implies that an observed structure would be expected to arise primarily from the action of social forces, versus by merit of the model permitting a large number of high probability structures, of which the observed structure is one (i.e. entropic effects). We also use our approach to identify the dyads within a given structure that are the least stable, and hence predicted to have the highest probability of changing under the current social forces. The utility of the "stable cone" for ERGM parameter optimization is then demonstrated on a physical model of amyloid fibril formation. This demonstration features a visualization of the stable region, whereby it is shown that the majority of the region of the model's parameter space where ERGM simulations produce the highest fibril yield lies within the "stable cone."

[*]This work was supported by NSF awards DMS-1361425 and SES-1826589.

[§]Authors Yue Yu and Gianmarc Grazioli contributed equally to this work.

[¶]yuey6@uci.edu

[‖]gianmarc.grazioli@sjsu.edu

[**]nolanephillips@fas.harvard.edu

[††]buttsc@uci.edu

[‡‡]To whom correspondence should be addressed: buttsc@uci.edu

Notation

We denote vectors by lower case boldface letters, as in $\mathbf{t}$, matrices by uppercase boldface letter, as in $\mathbf{M}$. $|S|$ is the cardinality of a set S.

1 Introduction

The last several decades have seen a growth of interest in network data and in strategies to analyze such data. While statistical analysis of social networks goes back to the 1930s [1], advances in computing and statistical theory have fueled a particular rise in stochastic models for networks with complex structure [2]. The importance of capturing nontrivial aspects of network structure has been motivated by studies of phenomena such as information transmission [e.g. 3, 4, 5], systemic robustness [e.g. 6, 7, 8, 9], and disease transmission [e.g. 10, 11], all of which can be significantly impacted by features such as clustering and community or subgroup structure that are not readily reproduced by simple random graph models.

Another motivation for complex network models has been the elucidation of the connection between local and global aspects of network structure. For instance, the frequency distribution of triadic subgraphs strongly constrains higher-order structures like ranked clusters [12], and partnership concurrency is closely related to forward connectivity in time-varying networks [11]. Biases in subgraph frequencies are themselves directly related to the conditions under which the state of one edge depends upon another [13, 14, 15], creating a direct link between local processes that e.g. favor or inhibit tie formation or triadic closure and higher-order structure. The exponential family random graph modeling (ERGM) framework (discussed in detail below) has become a widely used approach for identifying and exploring such connections between local and global structure in social and other networks [16, 17]. While it can be seen simply as a flexible language for specifying distributions on graph sets, ERGMs can also be interpreted as parameterizing a set of biases influencing relational structure, with realized networks emerging from the interplay of these biases; these biases are formally analogous to forces in a physical context, an analogy that has been exploited in applications of ERGMs to biophysical systems [e.g. 18, 19]. In some cases, these biases can also be interpreted in terms of utility theory [20], with network structure arising from the equilibrium of a latent stochastic choice process in which agents' decisions to add or remove ties are shaped by the associated biases. One potential use of ERGMs is hence to probe the conditions that are sufficient for the emergence or persistence of particular types of network structure, particularly where multiple mechanisms may be simultaneously at work.

In this paper, we examine one facet of this latter question, specifically introducing a basic notion of local *network stability* vis-á-vis an ERGM family and characterizing the subspace of parameters for an arbitrary family that renders a target structure stable with respect to a set of alternative networks. As we show, the stabilizing region of the parameter space forms the interior of a convex cone originating at the origin, whose faces are associated with a subset of the alternative networks against which the target is being compared. These results are presented in section 2, along with a practical algorithm for efficiently finding the stabilizing region of the parameter space. Our stability analysis is then illustrated with a simple and intuitive example involving an idealized centralized group structure (section 3), followed by an application to a well-known study of collaboration within a legal firm (in section 4), and finally (in section 5) the stable region for an ERGM used to model amyloid fibril formation (protein aggregates well-known for their connection to Alzheimer's disease). In the first two cases, the correspondence between our notion of local stability and persistence of structures under Metropolis dynamics (widely employed in Markov Chain Monte Carlo simulations of network structure) is explored, and the use of stability calculations to predict likely or unlikely edge changes is demonstrated; the third case further highlights the utility of local stability for aiding model selection. As these examples illustrate, the results described here are broadly applicable to ERGMs on social, biological, physical, or other networks, or *mutatis mutandis* to other discrete exponential family distributions on binary vectors.

2 Stability

We begin by introducing a simple notion of *local stability* for network structures. The intuition is as follows. Assume we have a graph whose stability is to be assessed (the *target network* or *target graph*) with respect to a set of *alternative graphs* that might be observed, as well as a model family whose elements are probability distributions on a graph set that includes the target graph and the alternative set. Our goal is to find a subset of models within the model family under which the target graph is more probable than any of the graphs in the alternative set. When this subset is non-empty, its members are said to *stabilize* the target graph vis á vis the alternative set, and by turns the target graph is said to be *locally stable* vis á vis the alternative set under any model within the subset.

It should be noted that our approach is very general: the target graph is given to us from the outset; there is no limitation on the choice of the alternative set; and although parts of our development require the model to have some specific functional properties, our core notion of stability is not limited to only the exponential family models. However, model families in ERGM form admit a particularly elegant characterization of stability, and we employ this framework throughout this paper.

In the remainder of this section, we formally define our notion of stability, and show how the use of the ERGM formalism allows the stabilizing subset of models within a given family to be easily characterized. We discuss some important properties of this stabilizing subset (including its geometric representation), and then provide an efficient approach to computing it in practice.

2.1 Definitions

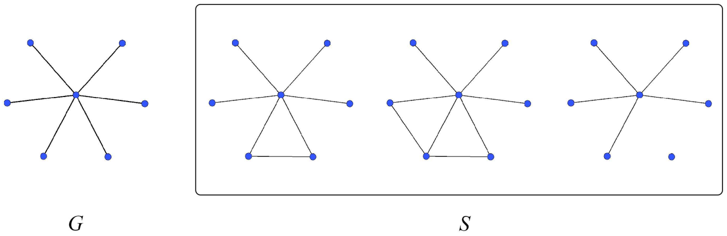


Figure 1: An example of a target graph G and alternative set S.

Let $\mathcal{G}$ be a finite set of graphs, $G \in \mathcal{G}$ be a *target graph* whose stability is to be assessed, and $S \subseteq \mathcal{G} \setminus G$ an *alternative set* of graphs with respect to which G is intended to be stable. For clarity of illustration we will emphasize the case in which all graphs in $G \cup S$ share the same vertex set, although this is not assumed. Likewise, we illustrate our ideas on simple graphs, but our development applies equally to directed, weighted, or multiplex networks.

We assess stability with respect to a model family on $\mathcal{G}$, which we take to be specified in ERGM form:

$$\Pr(G = g|\boldsymbol{\theta}) = \frac{\exp\left(\boldsymbol{\theta} \cdot \mathbf{t}(g)\right)}{\mathcal{K}(\boldsymbol{\theta})}, \text{ where } \mathcal{K}(\boldsymbol{\theta}) = \sum_{g \in \mathcal{G}} e^{\boldsymbol{\theta}^T \mathbf{t}(g)}, \tag{1}$$

where $\boldsymbol{\theta} \in \mathbb{R}^K$ is a vector of parameters with cardinality K and $\mathbf{t} : \mathcal{G} \mapsto \mathbb{R}^K$ is a vector of sufficient statistics. Intuitively, each element of $\mathbf{t}$ corresponds to a graph property whose distribution in G is to be biased, with the corresponding element of $\boldsymbol{\theta}$ indicating the strength and direction of the bias in question; in particular, $\mathbf{E}_{\boldsymbol{\theta}} t_i(G)$ is monotone in $\boldsymbol{\theta}_i$. Our goal is then to find the set of coefficients $\Phi = \{\boldsymbol{\theta}\}$ that make G more probable than any of the graphs in S, i.e.

$$\boldsymbol{\theta}, \text{ such that } \Pr(G|\boldsymbol{\theta}) > \Pr(G'|\boldsymbol{\theta}), \text{ for } G' \in S \tag{2}$$

with $\Phi = \{\varnothing\}$ if no $\boldsymbol{\theta}$ satisfies this condition. Any $\boldsymbol{\theta} \in \Phi$ is said to *stabilize* G with respect to S, and Φ defines the *stabilizing subfamily* of the model family parameterized by $\mathbf{t}$ for G with respect to S. For convenience, we also refer to Φ as the *stabilizing subspace* of the full parameter space, noting the equivalence of models and parameter vectors in this representation.

2.1.1 Characterization of the Stable Subspace

Using (1), we can rewrite the inequality in (2) as

$$\frac{e^{\boldsymbol{\theta}^T \mathbf{t}(G)}}{\mathcal{K}(\boldsymbol{\theta})} > \frac{e^{\boldsymbol{\theta}^T \mathbf{t}(G')}}{\mathcal{K}(\boldsymbol{\theta})}$$

allowing us to simplify the stability condition as

$$\boldsymbol{\theta} : \boldsymbol{\theta}^T (\mathbf{t}(G') - \mathbf{t}(G)) < \mathbf{0} \text{ for } G' \in S. \tag{3}$$

The quantity $\boldsymbol{\theta}^T \mathbf{t}(G)$ is called the ERGM *potential*, and is equal to the log probability of G up to an additive constant. $\boldsymbol{\Delta}(G, G') = \mathbf{t}(G') - \mathbf{t}(G)$ is called the *change score*, and describes the way in which the sufficient statistics differ between graphs. We may then construct a matrix $\mathbf{M}$ by accumulating the change score vectors for G versus all G' in S. Specifically, we define $\mathbf{M}$ to be a $|S| \times K$ matrix where the i-th row is the change score $\boldsymbol{\Delta}(G, G'_i)$, and the j-th column is the change scores regarding the j-th sufficient statistics:

$$M_{ij} = \Delta_j(G, G'_i), G'_i \in S.$$

With this notation it becomes clear that we can easily describe the local stability problem algebraically as

$$\begin{aligned} &\text{Find all } \boldsymbol{\theta} \\ &\text{Such that } \mathbf{M}\boldsymbol{\theta} = \mathbf{v} \in \mathbb{R}_-^{|S|} \end{aligned} \tag{4}$$

with the set of $\boldsymbol{\theta} \in \mathbb{R}^K$ that satisfy the above constraint defined as Φ.

The algebraic characterization of Φ immediately reveals several useful properties of the stabilizing subspace. Trivially, the matrix product $\mathbf{M}\boldsymbol{\theta}$ in Eq. 4 can be rewritten as a set of row-wise inner products arising from members of the comparison set:

$$\sum_j M_{ij}\theta_j < 0, \text{ for } 1 < i < |S|.$$

Each corresponding inequality represents an *open half space*, whose *dividing hyperplane* (eg. $\sum_j M_{ij}\theta_j = 0$) passes through the origin. Intuitively, each half-space represents the portion of the parameter space under which G is more probable than a particular member of S. The intersection of these open half-spaces, if nonempty, is the interior of a convex polytope cone emanating from the origin (Fig. 2). We call this convex cone the *stable cone* of G w.r.t the alternative set S, since any parameter vector within it stabilizes G.

We can also prove the set Φ is a convex cone by showing that if $\theta_1, \theta_2 \in \Phi$ then $\alpha\theta_1 + \beta\theta_2 \in \Phi$, for any α and $\beta > 0$. *Proof*: Let $v_1 = M\theta_1$ and $v_2 = M\theta_2$, then $v_1, v_2 \in \mathbb{R}_-^{|S|}$. So,

$$\begin{aligned} v' &= M[\alpha\theta_1 + \beta\theta_2] \\ &= \alpha v_1 + \beta v_2 && \text{because } \alpha, \beta > 0 \\ &\in \mathbb{R}_-^{|S|} \\ &\square \end{aligned}$$

A practical implication of this observation is that the stabilizing subspace can be characterized in terms of a set of vectors (the directions of the facial intersections of the stable cone), from which any member of Φ can be obtained. In practice, we might expect that redundancies in the constraints implied by M will limit the number of vectors that are needed, an idea that we exploit below.

We note in passing that the criterion used in Eq. 3 for defining stability implies that the margin of stability for G versus alternatives G' may be arbitrarily small. Replacing 0 with the log of the desired probability ratio in Eq. 3 loads to a larger guaranteed margin of stability. However, we can immediately perceive that the effect of such a margin change is simply to shift each dividing hyperplane towards the interior of the stable region by an amount that is logarithmic in the probability ratio. Equivalently, we observe that the margin of stability versus a given alternative increases exponentially with distance from its dividing hyperplane; as we show in Section 3.1, this in practice makes the seemingly weak, "zero-margin" stable cone a generically useful proxy for stronger criteria. We also note that, while the criterion of Eq. 3 is automatically applicable to exponential families, the condition can also be satisfied approximately or exactly by other families of distributions (particularly in the context of specific choices of G and S). Although we restrict attention to the exponential family case, the development used here may thus be applied in some other cases we asll.

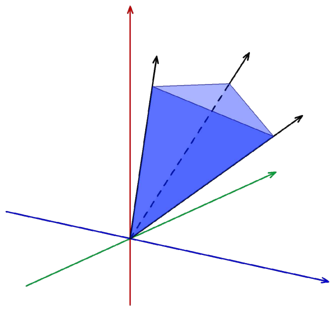


Figure 2: An example of the convex cone formed by 3 hyperplanes, in three-dimensional space.

2.2 Local Stability

The above stability definition can be applied to any arbitrary set of alternative graphs up to and including $S = \mathcal{G} \setminus G$; in the latter case, stability of G under $\boldsymbol{\theta}$ is obviously synonymous with G being the mode of the model specified by $\boldsymbol{\theta}$. Typically, however, we are interested in evaluating the stability of G with respect to a set of alternatives that are "close" to G under a hypothetical change process (e.g., the stochastic choice process of Snijders [20], the reaction kinetics of Grazioli et al. [18], or the Metropolis dynamics frequently used in MCMC algorithms [21]) in which the network of interest evolves via discrete changes in which single edges are added or removed from the graph (*dyad toggles*) such that "uphill moves" on the probability surface occur at a higher rate than "downhill moves." Here, we discuss a specification of S that is broadly useful for assessing local stability in such settings. The intuition is as follows. Let S be the set of all graphs in $\mathcal{G}$ reachable from G by a single dyad toggle (i.e., the Hamming sphere of radius 1 centered on G). When G is stable with respect to S, moves away from G will be disfavored (and return moves will be at least somewhat favorable); by contrast, when G is unstable with respect to S, there will be favorable dyad toggles that move away from G (with the return move being unfavorable). This notion of local stability (equivalent to G being a local mode in the Hamming space on $\mathcal{G}$) generalizes naturally to higher Hamming radii, and is a natural and easily computed starting point for considering longer trajectories (see fig.3 and fig.4).

Specifically, let $H^{(d)}$ be a set of all graphs that are Hamming distance d away from the target graph G. G may be said to be locally stable at (Hamming) radius d if G is stable with respect to $S = H^{(d)}$. In the special case of a single edge change, we are interested in $d = 1$. Solving for the stable cone (if exists) gives us the region within which the target graph lies at the local potential mode. If the stable region does not exist, it indicates that the target graph cannot be stabilized by the current choice of forces parameterized by $\mathbf{t}$, hinting that the model could be poorly designed.

Since the number of rows in M is the cardinality of S, the stable cone for a simple graph with v vertices arises from the intersection of $\frac{1}{2}v(v-1)$ half-spaces. Although this quadratic scaling (more generally, $\mathcal{O}(v^{2d})$) is unfavorable, it is typically the case that symmetries associated with $\mathbf{t}$ lead many rows of M to be identical; such redundant rows can be removed without changing the solution to the stability problem. In some cases, the resulting compression can be considerable: for instance, one of the examples we show in section 3.1 leads to a two-row M matrix regardless of v. Moreover, some unique rows of M may also be redundant, in that they specify constraints on the stable region that are weaker than the constraints imposed by the other rows of M. As we show below, this form of redundancy can be exploited to calculate the stable cone in practical settings (as illustrated in section 4).

2.3 Solving for the Stable Cone

We employ the following terminology to refer to geometric features associated with the stability problem. Assuming we are in a K-dimensional parameter space, a $K - 1$-dimensional dividing hyperplane separates the region of the parameter space where the target graph is more probable than a specific alternative graph $G' \in S$ from the region of the parameter space where the target graph is at most equiprobable to G'. A dividing hyperplane may or may not constrain the stable cone (the intersection of half-spaces imposed by all dividing hyperplanes associated with the rows of M). When it does, the "side" of the cone created by its

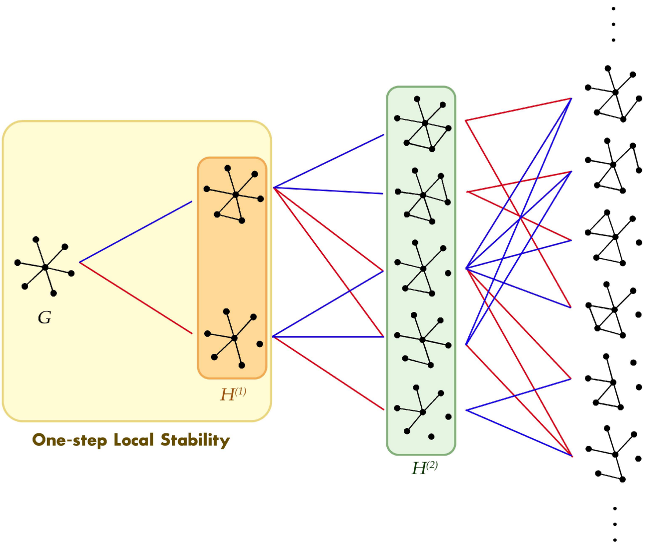


Figure 3: Examples of Hamming trajectories that could be taken from a target graph G. Here we group all isomorphic graphs into the same representation, and we only consider trajectories without loops. Each step is shown as either a blue line or a red line. Blue lines indicate steps involving edge addition and red lines indicate edge deletion. In this example, the set $H^{(1)}$ only contains two graphs, and local stability is satisfied if moves to either of the two graphs are disfavored.

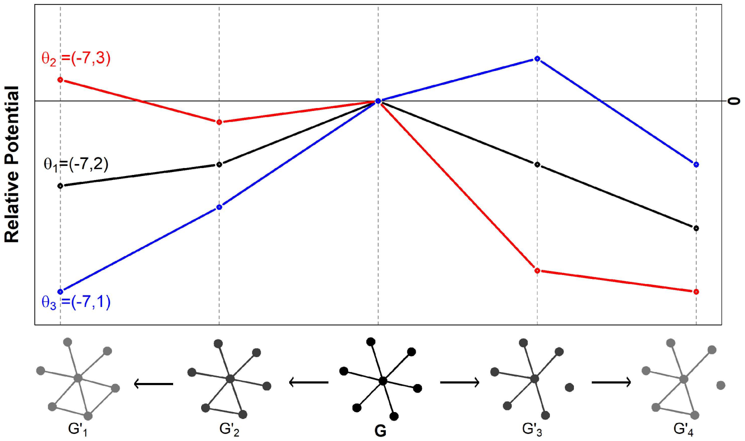


Figure 4: Illustration of a local mode in Hamming space. We choose two example trajectories, both started from the target graph G (plotted in the center). Trajectory G-G'_2-G'_1 is plotted from G to left and trajectory G-G'_3-G'_4 is plotted from G to right. Relative potential is the potential gain moving away from G. The model here has two terms: `edges` and `kstar(2)`. We choose three different θ's, and the relative potentials for each choice of θ are plotted in different colors. Under the model with coefficients θ_1 (in black), G has highest potential among these five graphs. In fact, G has the highest potential among all graphs in $H^{(1)}$ and $H^{(2)}$, indicating that G is stable against any two-step toggles. Under θ_2 (in red), G is *locally stable* because it has higher potential than both graphs in $H^{(1)}$. Note that although G'_1 has higher potential than G under the coefficients θ_2, trajectory to G'_1 is separated by graph G'_2 whose potential is lower than G. Under the coefficients θ_3 (in blue), G is considered unstable, due to the fact that moves to G'_3 increase the potential. In fact, coefficient θ_3 stabilizes G'_3 instead.

intersection with the hyperplane is called a *facet*. An intersection of two facets is called a *ridge*, which is an element of dimension $K - 2$.

The stable cone can be characterized using two different geometric representations: the *halfspace representation* (or *H-representation*) and the *vertex representation* (or *V-representation*) [22]. The H-representation characterizes the subspace Φ as a set of conditional linear inequalities, or half-spaces, as shown by the blue facets in Fig. 2, and in Eq. 5 below:

$$\Phi = (\theta | M_i \theta < 0) \ \forall i \in \{1, 2, ..., |S|\} \tag{5}$$

while the V-representation characterizes the subspace as the convex hull generated by the vertices that are created by each pair of intersecting planes from the H-representation (shown as black arrows in Fig. 2). It is worth noting that, while solving for the stable region, it is convenient to store each vertex as a single point in parameter space by calculating the intersection of each vertex with a hypersphere of a given radius (shown in Fig. 5). Since all points along the ray associated with each vertex can be obtained by rescaling, as can the ray associated with any point within this particular slice of the stable region (solid orange triangle in 5), no information is lost by using this representation. Additionally, the data structure for the V-representation is such that the indices of the hyperplanes whose intersection comprise each vertex is also stored in the V-representation object. The remainder of this section presents the calculation of the stable region using this method for storing the V-representation.

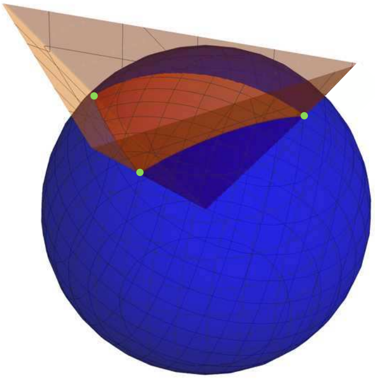


Figure 5: The vertex representation (intersections of the transparent orange planes) can be conveniently stored as the points (green dots) produced by the intersection of the vertices with a hypersphere (blue sphere)

Although the H and V representations each offer a complete description of the stable cone individually, they are perhaps most useful for solving the stable cone when the two representations are used in tandem. In our application, we leverage the complementarity of the two representations to iteratively search for both a non-redundant H-representation and V-representation simultaneously by applying a methodology based on the *double description method* (or *DD method*), introduced by Avis and Fukuda [22]. Given an initial redundant $\mathbf{M}$ matrix (*redundant halfspaces*), with all edge changes of interest represented in the rows, the stable cone for a given target graph and set of sufficient statistics can be solved by first using Algorithm 1 (Fig. 6) to obtain an initial closed superset of the stable region, which is then passed (along with $\mathbf{M}$) to Algorithm 2 (Fig. 7), where the initial closed superset is whittled down to a non-redundant double description of the stable region. For the following demonstrations of Algorithm 1 and Algorithm 2, consider a heuristic parameter space θ, for which $|\theta| = 3$, where we will examine the intersection of the stable region with a sphere. To further aid spatial intuition, the stable region displayed in Figures 6 and 7 is analogous to examining only the spherical triangle shown in opaque orange in Figure 5.

Given the redundant $\mathbf{M}$ matrix for a target graph and a set of sufficient statistics, this information is first passed to Algorithm 1 in order to calculate an initial closed superset of the stable region. Algorithm 1 begins by first obtaining a set of $|\theta| - 1$ rows from $\mathbf{M}$, where no two halfspaces are parallel, and calculating their intersection with both each other and the hypersphere. These rows and this vertex point serve to initialize the H-representation (H) and V-representation (V), respectively. At this point, a new row is drawn from $\mathbf{M}$ and appended to H, and intersections between all possible combinations of $|\theta| - 1$ rows of H are calculated and then appended to V. Although the current iteration of H is redundant, its stable region is still equivalent to the non-redundant form, thus H is used to remove all unstable vertices (i.e. vertices positioned outside the stable region) from V using matrix multiplication by H to evaluate the stability of each vertex. Given that the data structure for V includes the labels for the rows of H that intersect to form each vertex in V, it is then trivial to now remove all rows of H that are not represented in V. At this point, H and V are both non-redundant. The final step in the loop is to check the convex hull for closure. If the convex hull represented by H and V is closed, Algorithm 1 returns H and V and terminates. One straightforward method for testing the closure of the convex hull described by V is to first check that the number of vertices is $\geq |\theta|$, and then use any of the many available methods for calculating convex hulls from points (e.g. Quickhull [23]) to calculate the convex hull of the vertices in V. If both the number of vertices is $\geq |\theta|$, and the number of halfspaces in the halfspace representation returned by the convex hull finding method is equivalent to the number of rows in H, the stable region represented by H and V is closed. This test for closure is made possible by the fact that if the convex hull finding algorithm is operating on a convex hull that is open with respect to the stable region, it will introduce a new halfspace to close off the open end of the space.

Algorithm 1: Finding an initial closed superset of the stable region

```
Data: $\mathbf{M}$
initialize H; V;
bool are.parallel = TRUE;
while are.parallel do
    h.init = sample.two.rows($\mathbf{M}$);
    are.parallel = check.parallel(h.init)
end
H = h.init;
V = get.exhaustive.intersections(H);
bool hull.is.closed = FALSE;
while !hull.is.closed do
    h.test = sample.one.row($\mathbf{M}$);
    H = append(H, h.test);
    vertices.new = get.exhaustive.intersections(H, h.test);
    V = append(V, vertices.new);
    V = return.stable.vertices(V, H);
    H = get.H.from.V(V);
    hull.is.closed = testForClosedConvexHull(V);
    if(h.row.index == nrow($\mathbf{M}$)) break;
end
return {H, V}
```

Once Algorithm 1 has returned an initial closed superset of the stable region, H, V, and $\mathbf{M}$ are then passed to Algorithm 2. The premise of Algorithm 2, as put forth by [22], is that given an initial closed convex hull, any newly introduced halfspaces are only non-redundant if their introduction excludes one or more previously existing vertices. The algorithm also leverages the fact that the data structure for V keeps track of which halfspaces intersect to produce each vertex, in that upon introduction of a new non-redundant halfspace, only the halfspaces whose intersections comprise the newly excluded vertex or vertices must be included in the calculation of newly created vertices. Intuition for the methodology can be readily obtained from Fig. 7.

```
Algorithm 2: Using the DD method to solve the stable cone
Data: H.initial, V.initial, M
initialize H = H.initial; V = V.initial;
for all rows in M do
    h.test = get.next.row(M);
    v.excluded = get.excluded.vertices(V, h.test);
    if is.not.empty(v.excluded) then
        H = append.row(H, h.test);
        rows.for.testing = get.rows.comprising.vertices(v.excluded);
        vertices.for.testing = get.new.intersections(rows.for.testing, h.test);
        v.new = get.stable.vertices(vertices.for.testing);
        V = append(V, v.new)
end
return {H, V}
```

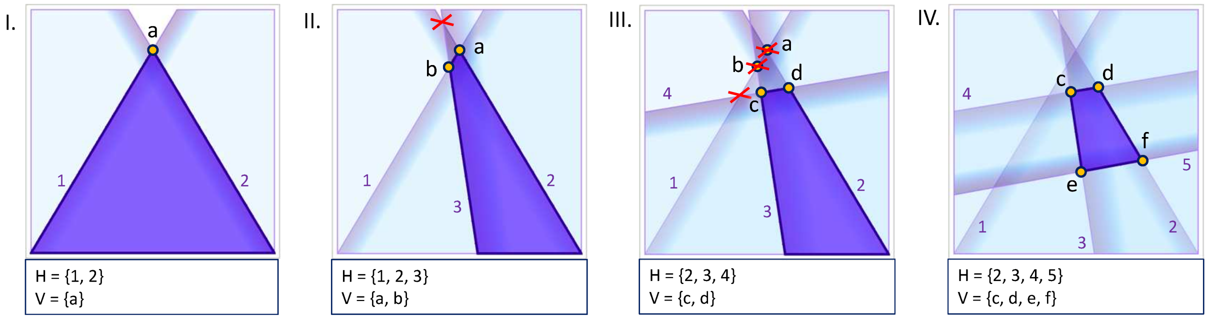
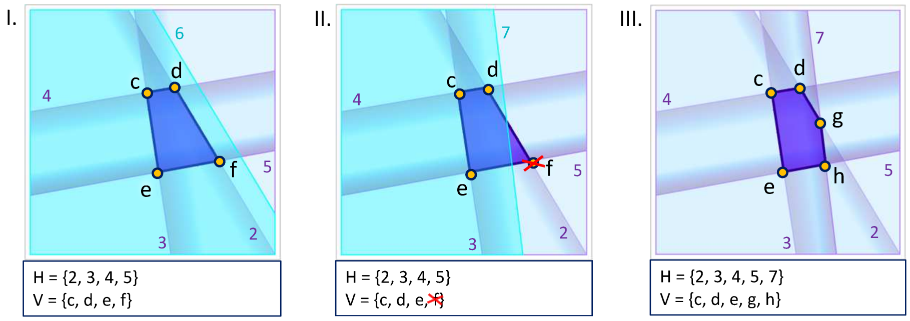


Figure 6: A demonstration of Algorithm I, used to define an initial closed superset of the stable region. I.) The first step in Algorithm I: the H-representation is initialized with two non-parallel halfspaces drawn from **M**, while the V-representation is initialized as the intersection of the two halfspaces in H. II.) Step 2 in algorithm I: A third halfspace is introduced, all n choose 2 intersections are calculated, all intersections within the stable region defined by H become the V-representation, and all halfspaces whose intersections comprise V become H. The convex hull is not closed, so we iterate another step. III.) Step 3 in algorithm I: as in the previous step, a new halfspace is introduced, stable intersections become V, and their respective halfspaces become H. Note that in this case, two previously included vertices (a and b) are now excluded from V, as is halfspace 1, since it no longer contributes any stable intersections to V. Still, the convex hull is not closed, so we iterate another step. IV.) A new halfspace is introduced, and the process from the previous two steps is repeated. This time, the convex hull is closed, thus Algorithm 1 is terminated, and an initial closed superset of the stable set is returned.

The order of computational complexity for this methodology is best understood using a bounding argument. First, we establish the two fundamental operations employed by our methodology: 1) calculating intersections of sets of halfspaces that form vertices, 2) testing whether early iterations of the stable region are closed, and 3) testing whether or not vertices lie within the stable region. Since step 2 is carried out minimally during early iterations, we begin by first addressing steps 1 and 3. The first operation is an inversion of a $K \times K$ matrix, where K is the dimensionality of the model (i.e. the number of sufficient statistics), and the third operation is a simple matrix multiplication of a vector of length K by the matrix representing the current iteration of the H-representation (with trivial parallelization); thus, between the first and third steps, the rate-limiting operation is the calculation of vertices. Next we establish the scaling for a brute force treatment whereby, first, all n choose $K-1$ intersections between hyperplanes are taken to obtain all possible vertices (operation 1), then each vertex is tested to determine whether or not it is within the stable region (operation 2), leaving the non-redundant double description. In this case, the computational complexity scales as $\mathcal{O}((\frac{en}{K})^K)$. The worst-case scenario for our methodology would be that *Algorithm 1* fails to produce a closed stable region until the very last row from the M matrix has been included in the exhaustive search, demonstrating that, at its worst, the order of complexity for our algorithm is equal to the



H = {2, 3, 4, 5}
V = {c, d, e, f}

H = {2, 3, 4, 5}
V = {c, d, e, ~~f~~}

H = {2, 3, 4, 5, 7}
V = {c, d, e, g, h}

Figure 7: A demonstration of Algorithm II, which takes an initial closed superset of the stable region and an **M** matrix as input, and outputs H and V descriptions of the stable region for a given model. I.) The closed set from figure 6 (step IV) becomes the input for Algorithm II.) A new halfspace is introduced (cyan), but since no vertices from V are excluded, the halfspace is redundant, and thus rejected. II. Another halfspace is introduced, which excludes vertex f, thus f must be rejected from V and all intersections between the new halfspace and those whose intersections comprise f must be calculated and tested for stability under H. III.) The new halfspace is appended to H and the intersections between 2, 5, and 7, stable under H are appended to V.

exhaustive brute force treatment. Next, consider the best-case scenario: given that a particular stable region can be expressed as an H-representation comprised of h rows, suppose that these are the first h rows selected from the M matrix in *Algorithm 1*. In this case, the only vertices calculated would be those produced by the H-representation, implying a computational complexity of $\mathcal{O}((\frac{h}{K})^K)$. Even for a pessimistic case where only half of the rows in M are redundant ($h = nrow(M)/2$), our methodology would offer a computational speedup of $.5^{-K}$, a massive speedup over the fully exhaustive brute force method (e.g. 32× speedup for a network model with just 5 sufficient statistics). Although step 2, testing for closure, is only carried out during early iterations and is not a rate-limiting step, for the sake of completeness, we next address the order complexity for step 2. The test for closure is carried out by determining whether the graph of halfspaces (nodes) sharing vertices (edges indicate a shared vertex) contains a k-cycle of size $k \geq D$, where D is the number of dimensions in parameter space. Consequently, the computational complexity of this step will scale $\mathcal{O}(VE)$, where V and E are the vertices and edges in the aforementioned graph describing the connectivity be-tween halfpaces and vertices in the double description of the stable region [24].

3 Case Study I: Cult (Star) Structure

In this section, we apply the above method to a simple example of a social structure that we here refer to as the *cult network*. A cult network is characterized by an isolated star structure, where there is a single core ("leader") node in the center connecting to all peripheral ("follower") nodes, and no peripheral nodes are mutually adjacent. Thus, a follower can reach any other follower, but only via a path that is brokered by the leader. Although the motivation behind introducing our methodology using a cult network as an example is illustrative simplicity, it does represent a stylized version of features found in some real-world charismatic cults. For example, both the People's Temple [25] and Heaven's Gate [26] cults featured leaders who eventually isolated their groups from outside contact and regulated the flow of critical information such that rank-and-file members were encouraged to trust and obey only the leaders themselves (with unmediated intra-group relationships among members being heavily discouraged). Maintaining these star-like group structures required considerable creativity and effort on the parts of the cult leaders, who ultimately crafted complex social environments that stabilized what would otherwise be highly unfavorable pattern of social relationships. The cult network model presented here demonstrates the conditions that are necessary for such a structure to be stabilized under one very simple class of social processes.

The model family employed here is minimal by design, incorporating only two types of social "forces:" a general propensity to form edges, and a force that governs the propensity for untied pairs of individuals to have or lack partners in common. In ERGM form, such a family corresponds to a vector of statistics ($\mathbf{t}$) containing respectively the count of edges (`edges`) and the count of null dyads (i.e. pairs of vertices that do not share an edge) having no partners in common (nsp(0)). We investigate whether the star structure can be stabilized under this model family, and if so, define the region of the parameter space where that the network is stable.

In the remainder of this section, we first compute the stable cone, then demonstrate that this region aligns with stability as assessed by simulated Metropolis dynamics. We also consider the question of which dyad is likely to be the first to be toggled (assuming that some toggle occurs). This approach can help identify which edges/nulls are most vulnerable to change under the model. We can also show analytically the probability of the target network transitioning to each of the alternative networks, given that an edge change has occurred. We then validate these results with simulated trajectories.

We use an undirected star network of v vertices as the target network. For illustrative purposes, we let $v = 7$ when figures or numeric results are generated (e.g., G in figure 8). For simplicity, we assume that no attribute is associated with vertices or edges, indicating that all peripheral nodes are interchangeable. Because our model is unable to differentiate between the target graph and other members of its isomorphism class, all isomorphic graphs will be counted towards stability. The set $S = H^{(1)}$ contains only two distinct networks (illustrated in fig. 8): G_+ is the result of adding one edge to G, connecting two peripheral nodes; and G_- is the result of removing one edge from G, breaking one connection between the core and a peripheral node. As mentioned, the two sufficient statistics are edges and nsp(0)). We note in passing that there are other model families that can also generate star structures. For instance, a term influencing the number of null dyads having exactly one shared partner (nsp(1)) can also be used to generate star-like graphs. However, the nsp(0) term, despite being less obviously associated with the star structure, interacts with the edges term to anchor the structure in place, i.e. a negative value in nsp(0) suppresses the existence of non-edges sharing exactly zero partners, thus non-edge pairs with higher shared partners are boosted, to a limit that is controlled by the network density (set by edges).

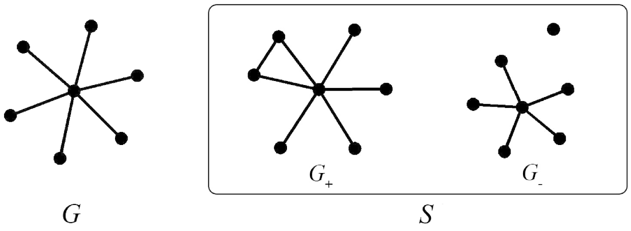


Figure 8: $S = H^{(1)}$ for the star structure.

3.1 Stable Parameter Region of the Star Structure

The rows of the M matrix are comprised of the change scores (i.e., differences of graph statistics) between the target graph and the graphs in the alternative set (shown in table 1). The parameter space of this model is 2-dimensional - each sufficient statistic of the model occupies one dimension. To illustrate the parameter space we plot edges on the horizontal dimension (x-axis) and nsp(0) on the vertical dimension (y-axis). For simplicity, hereafter we use x, y to represent the parameter values of edges and nsp(0), respectively.

	Sufficient Statistics (t)		Change Score (M)	
	edges	nsp(0)	edges	nsp(0)
G	v-1	0	-	-
G_+	v	0	1	0
G_-	v-2	v-1	-1	v-1

Table 1: Sufficient statistics and change scores for the star structure G and the set $S = H^{(1)}$. v is the number of vertices in the graph; edges and nsp(0) are the two ERGM terms used in the model.

As derived in section 2.1.1, each graph in S defines a half-plane (or more generally, a half-space; in subsequent discussions, we use general terms such as half-spaces and hyperplanes, despite them being called half-planes and lines in 2-dimensional space): $G+$ defines $x \leq 0$, and G_- defines $x - (v-1)y \geq 0$. The stable region, which is the intersection of the two half-spaces, is plotted as the shaded area in fig. 9a (assume $v = 7$). As proved in section 2.1.1, the stable region is a cone that has infinite height and points at the origin.

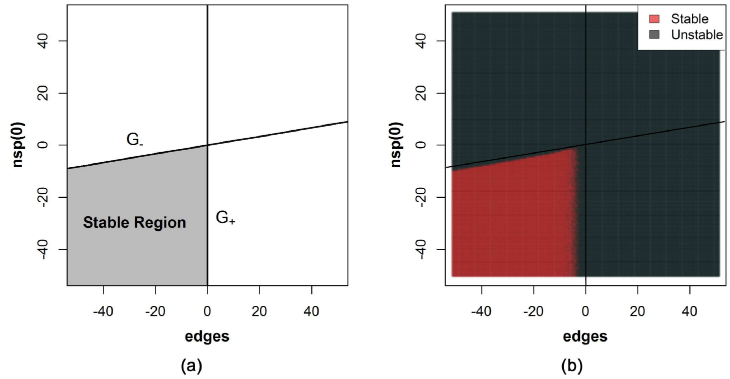


Figure 9: **a**. The stable cone (grey region) for the star structure under the `edges/nsp0` family, assuming $v = 7$. **b**. Whether simulated networks remain in star configurations after 10^6 random-walk Metropolis steps for grid-selected parameter values (dots). The points are sampled at equal distance apart (0.4 in each axis) from -40 to 40. The color values range from grey (no persistence) to red (complete persistence). Despite being defined only relative to $H^{(1)}$, the stable cone (line-bounded region in lower left) closely matches the region of long-term dynamic stability.

To verify that the stable cone is compatible with stability under explicit dynamics, we sample 10^6 parameter vectors and run 50 random walk Markov chains from each sampled vector. (All simulation was performed using the `ergm` package within `statnet` [27, 28] using random dyad proposals.) Each chain starts from the perfect star structure and runs for $t = 10^6$ Metropolis steps. We examine the persistence of the star structure in the networks at the end of the simulation, and plot the fraction of the networks that remain unchanged (fig. 9b).

Even though the stable cone is based only on stability versus $H^{(1)}$, it corresponds closely to long-term stability under Metropolis dynamics (red points in Fig. 9b). Outside of the stable cone, the graph is dynamically unstable (dark grey points in fig. 9b Along the interior of the two facets, there is a thin stripe of models that are dynamically unstable despite being inside the locally stable region. This is reflective of the fact that, while moves away from G are unfavorable, they will eventually happen given enough attempts (here, 10^6). Parameter values close to the faces of the stable cone have a lower margin of stability, in the sense that probability gap between G and the elements of S is smaller, and dynamic stability hence begins to be lost as one moves from the center of the locally stable cone to its faces. An interesting observation is that the "unstable band" associated with G_+ is a bit wider compared with the band associated with G_-. This phenomenon can be explained by examining the ERGM potential of each alternative graph. In this particular case, the difference between the width of the unstable bands for G_+ and G_- reflects differences in entropy. Since there are only 6 different ways to move from the target graph to G_- (removing one of the 6 edges), and 15 different ways (6 choose 2) to add an edge to the target graph to reach the G_+ state, there are more than twice as many ways for an MCMC sampler to move "downhill" in likelihood from the target graph to the state G_+.

By definition, the stable cone is the region where the ERGM potential of the target graph, $\boldsymbol{\theta}^T\mathbf{t}(G)$, is greater than any of the other graphs in S; facets and ridges of the cone then correspond to the sets of $\boldsymbol{\theta}$ values where at least one or two graphs (respectively) in S have the same potential as the target graph. Because the graph probability is proportional to the exponentiated ERGM potential, $\Pr(g) \propto \exp(\boldsymbol{\theta}^T\mathbf{t}(g))$, examining graph potentials provides insight into the behavior of the Metropolis dynamics. As shown in fig. 10, the potential difference of G and $G_+$is much more gradual along the dividing hyperplane compared to that of G and G_-, thus the down-potential step is taken with higher probability during the Markov Chain run.

The above simulation experiment considers whether network snapshots under a long simulation are isomorphic to the target graph. In principle, a network could alternate between a number of states during the

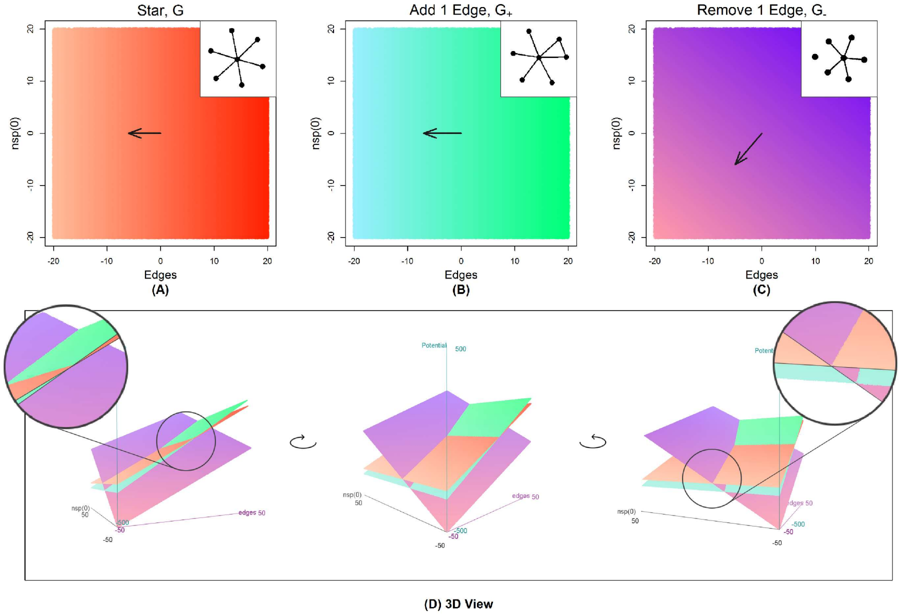


Figure 10: Potential planes of target graph G, and both alternative graphs G_+, G_- in S. Panels A, B, and C plot the potential of networks G, G_+ ,and G_-, respectively. The gradients of the potential surfaces are plotted as arrows. Panel D plots the three gradients onto the same space. Note that the stable cone can be identified in the bottom left quadrant, where the potential of G is higher than both G_+ and G_- (see inset).

simulation, counting towards stability so long as it is in the target state when observed. A more rigorous test of dynamic stability is to examine the expected time to the *first* change under Metropolis dynamics across the stable cone. Fig. 11 shows the mean number of steps required for the first accepted toggle, from yellow (1 step) to red ($>$ 1000 steps). As in the previous simulation, we see that the graph strongly resists perturbation within the bulk of the stable cone, staying unchanged for more than 1000 simulation steps. The wider stripe of dynamic instability along the $x = 0$ hyperplane ($P(G) = P(G_+)$) is associated with more rapid network changes, with fewer than ten simulation steps required on average as one nears the boundary of the stable cone. The relatively sharp transition from the less dynamically stable region to strong dynamic stability results from the exponential decline in acceptance probabilities with respect to the potential difference: while our definition of S leads to an extremely local definition of stability, the exponential decline in transition probabilities as one moves away from the faces of the stable cone makes it a good proxy for dynamic stability more broadly defined.

3.2 Dyad Vulnerability

We have shown that the target structure is most dynamically stable well within confines of the locally stable cone, with dynamic stability weakening in the region closer to the facets of the cone. Of course, given enough attempts, a change will inevitably occur; this raises the question of which structure in S will be selected when the first dyad toggle is accepted. This can be thought of as a problem of *dyad vulnerability*: if change happens, which dyads are most vulnerable to being toggled? In the case of the cult network, symmetry leaves us only two types of possible dyad toggles: toggles that break connections between the center vertex and one of the periphery vertices (denoted d_-, resulting in a graph isomorphic to G_-), or toggles that establish connections among periphery vertices (denoted d_+, resulting in a graph isomorphic to G_+).

Assume a dyad toggle d_i is selected randomly from all possible $\frac{1}{2}v(v-1)$ toggles. The two types of toggles are proposed at rates based on the frequencies they appear in G: $\Pr(d_-) = \frac{2}{v}$, and $\Pr(d_+) = \frac{v-2}{v}$. Then the acceptance ratio ($\alpha = \Pr(G_i)/\Pr(G)$) is calculated. The proposed toggle is accepted if the new graph is more probable, i.e., $\alpha > 1$. Otherwise, the toggle is accepted with probability $\Pr(\text{accept}|d_i) = \alpha =$

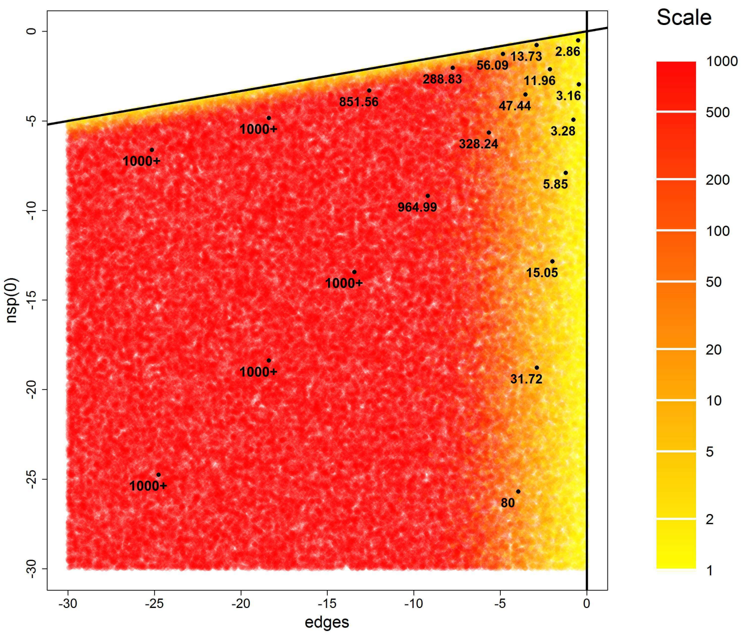


Figure 11: Mean number of Metropolis steps required for the first change in G, as a function of $\boldsymbol{\theta}$. Waiting times grow exponentially fast as one moves away from the faces of the stable cone.

$\exp(\theta^T(t(G_i) - t(G)))$. The acceptance probability if d_- is proposed is:

$$\Pr(\text{accept}|d_-, G) = \begin{cases} \exp(\boldsymbol{\theta}^T(t(G_-) - t(G))), & \text{if } \boldsymbol{\theta}^T(t(G_-) - t(G)) \leq 0 \\ 1, & \text{otherwise.} \end{cases}$$

Then the probability of the target graph G change to G_- can be calculated as

$$\Pr(G_-|G) = \Pr(d_-)\Pr(\text{accept}|d_-).$$

Similarly, $\Pr(G_+|G)$ can be derived when d_+ is proposed. When proposed moves are rejected, then the graph remains unchanged. Let $\boldsymbol{\theta} = [x, y]^T$ where x is the parameter value for `edges` and y is the parameter value for `nsp(0)`. The probability of becoming G_+, G_-, or staying unchanged given starting from the perfect star structure G is:

$$\Pr(G_-|G) = \begin{cases} \frac{2}{v}e^{-x+(v-1)y}, & \text{if } x - (v-1)y \geq 0 \\ \frac{2}{v}, & \text{otherwise} \end{cases},$$
$$\Pr(G_+|G) = \begin{cases} \frac{v-2}{v}e^{x}, & \text{if } x \leq 0 \\ \frac{v-2}{v}, & \text{otherwise} \end{cases},$$
$$\Pr(G|G) = 1 - \Pr(G_+|G) - \Pr(G_-|G)$$

To illustrate the dyad vulnerability, we sampled 10,000 parameter vectors from the $[-10, 10]$ interval for both `edges` and `nsp(0)` at $v = 7$. This region covers both the stable cone and the unstable region. We ran a 5000 step random walk Metropolis trajectory for each sampled parameter until the first change occurred. The Metropolis procedure evaluates the acceptance ratio (α) at each step and decides whether the proposed toggle is accepted or rejected (as mentioned in 3.2). The simulation result (in fig. 13) is in accordance with the theoretical derivation.

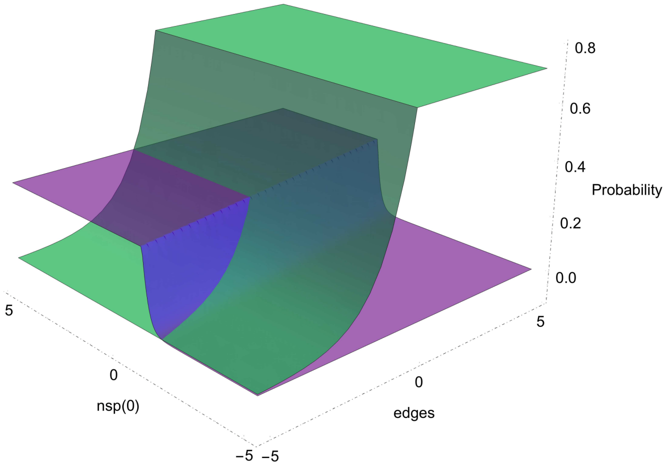


Figure 12: One step transition probability to each of the structures in $H^{(1)} = \{G_+, G_-\}$, as a function of the model parameters. The green surface corresponds to G_+, and the purple surface corresponds to G_-. The probability of staying unchanged as G is 1 minus the sum of the two surface values.

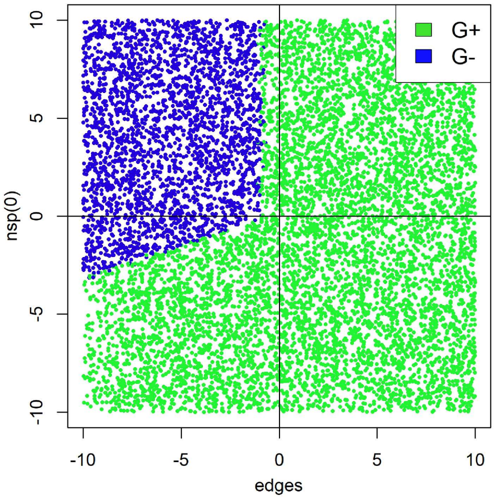


Figure 13: Simulation showing the most vulnerable dyad type, as a function of model parameters. Dots indicate sampled parameters, each of which was used to govern one MCMC trajectory; blue dots are networks that changed to G_- on the first toggle, indicating that breaking an existing connection is more likely, and green dots are networks that first changed to G_+, indicating establishing a connection between two periphery nodes are more likely.

4 Lazega's Lawyer Dataset

In this section, we analyze network stability with a real-world dataset and a published model. We note at the outset that although perfect local stability (i.e., the stable cone w.r.t. $H^{(1)}$ is non-empty) is reached with the star example, not all model families lead to non-empty regions of local stability for $S = H^{(1)}$. For instance, consider a network with an attribute that classifies nodes to be in either group A or B, and a model family that has two terms, a general `edges` term to control the overall density and a `nodemix` term that counts the number of between-group ties. If there exist vertices $(a, a') \subseteq A$ and $(b, b') \subseteq B$ such that edge $\{a, b\}$ is in G and edge $\{a', b'\}$ is not in G, then the change scores for toggling the $\{a, b\}$ and $\{a', b'\}$ dyads have respective elements `edges` = -1, `nodemix(A,B)` = -1 and `edges` = 1, `nodemix(A,B)` = 1. These two change score vectors define two half-spaces whose intersection is empty (i.e., $x + y > 0$ and $-x - y > 0$), and thus there is no model that stabilizes such a graph. Intuitively, this is because any choice of $\boldsymbol{\theta}$ for this family will on balance either favor adding or removing cross-group edges, and hence one of the two dyad states will not be favored. More generally, for any G and model family parameterized $\mathbf{t}$, if there exists an edge and a null in G with directly opposite change scores with respect to $\mathbf{t}$, then the local stable region associated with $S = H^{(1)}$ will be empty. In such cases, we can conclude that the forces associated with $\mathbf{t}$ do not (or would not) locally stabilize G. Beyond this observation, we can gain additional insight into model behavior by examining the subsets of S for which stabilization *is* possible, particularly where $\boldsymbol{\theta}$ is known or has been estimated from the observed network [see e.g. 21]. Moreover, where the local stable cone is non-empty but an estimated model does not lie within it, we can exploit the position of the estimated model relative to the stable cone to obtain insights into the factors that are driving instability, and the hypothetical changes in social forces that would lead the target graph to become locally stable. In this section, we illustrate how some of these techniques can be used to gain insights into model behavior in a non-trivial setting.

We study graph stability with a data set collected by Lazega [29] on working relations among 36 partners in 1991 in a New England corporate law firm. This dataset is a network where edges (undirected) represent collaborations between partners. For purposes of analysis, we employ a model family for this data set that was previously published by [30]. Due to improvements in estimation methods since the original publication, we here refit the model using the `ergm` package [31] to obtain updated coefficients (table 2). The covariates used in the model are as follows: seniority, which describes the rank order of entry into the firm (1=earliest, 36=latest); type of practice (1=litigation, 2=corporate); the office at which the partner works (1=Boston; 2=Hartford; 3=Providence); and the partner's gender (1=man; 2=woman). The model includes the main effects of seniority and practice, along with homophily effects for practice, gender, and office location. A geometrically weighted edgewise shared partner (GWESP) term was also included to account for triadic closure.

Parameter	Estimate	S.E
Edges	-7.375	0.712^{***}
Main Seniority	0.024	0.007^{***}
Main Practice	0.411	0.118^{***}
Homophily Practice	0.761	0.192^{***}
Homophily Gender	0.696	0.256^{**}
Homophily Office	1.145	0.196^{***}
GWESP($\alpha = 0.75$)	0.937	0.159^{***}
$^{*}\ p < 0.05$, $^{**}\ p < 0.01$, $^{***}\ p < 0.001$		

Table 2: Maximum likelihood estimates for the Lazega model, with standard errors and Wald test p-values. α refers to the GWESP decay parameter, which was fixed to 0.75 during estimation.

We enumerate $S = H^{(1)}$ (total 630 graphs) and compute the corresponding M matrix (630-by-7). Each row in M is characterized by one graph in the alternative set and each column is one model statistic. The network would be locally stable under the estimated model with parameter $\hat{\boldsymbol{\theta}}$ if $\mathbf{M}\hat{\boldsymbol{\theta}} \in \mathbb{R}_{-}$, i.e. if all 1-step changes are unfavorable. Interestingly, the observed graph is unstable against this alternative set with an unstable fraction of 0.159 (i.e., 100 out of 630 alternative graphs are unstable). Because each graph in the alternative set corresponds to one dyad toggle and defines a stable half-space, then for each dyad there exists a stable half-space in which this dyad is more likely to stay unchanged. If a model lies within the stable half-space, then the dyad is said to be stable under this model, and vice versa. We calculate the distance from the model to stabilization of graph G'_i in $H^{(1)}$ as *generalized distance* d_i (shown in fig. 14) and use the convention that positive d_i means the model lies outside of the stable half-space of graph G'_i; while negative d_i means the model is within the stable half-space of graph G'_i. The absolute value of d_i is the distance to the dividing hyperplane.

In fig. 15 we plot the stable edges, unstable edges, and unstable nulls. A few observations can be made: 1) Two core groups are clearly identified, each centered within one of the larger offices (Hartford and Boston). To illustrate the core groups, light shadings are plotted around the actors that have at least one stable tie within the group. The two groups are completely separated, which means the ties between groups are mostly

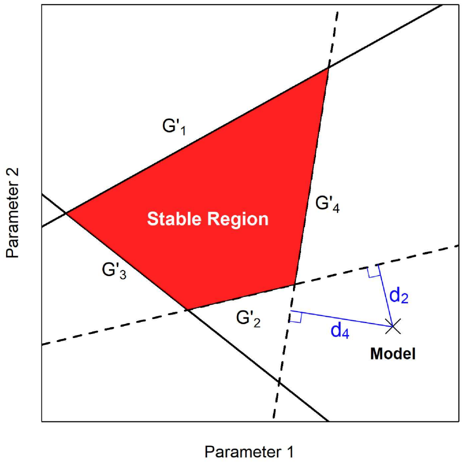


Figure 14: Example parameter space illustrating the stable region (in red), a particular model (black ×), and its distances to nearby hyperplanes (blue lines). Here, the target graph G is unstable under the model, as the model is positioned outside of the stable region. Nevertheless, the model is partially stable w.r.t. G'_1 and G'_3, because it lies on the side of the stable half-spaces of these two alternative graphs (bottom of G'_1 and left of G'_3). In order to quantify the instability of the model, one can calculate the generalized distance from the model's position in the parameter space to each of the hyperplanes bounding the halfspaces for which the model is not included.

unstable (in red). 2) Both large offices demonstrate core-periphery structures internally. The cores (marked by shaded background) are characterized with stable connections, while the peripheries are connected by unstable connections. 3) There are relatively few unstable nulls, and most of them are within two dense groups. In fact, there is only one unstable null that connects two groups. 4) Within two offices, there are 14 unstable nulls in the Boston office and only four in the Hartford office, suggesting that the pressure for forming new connections is less in the Hartford office.

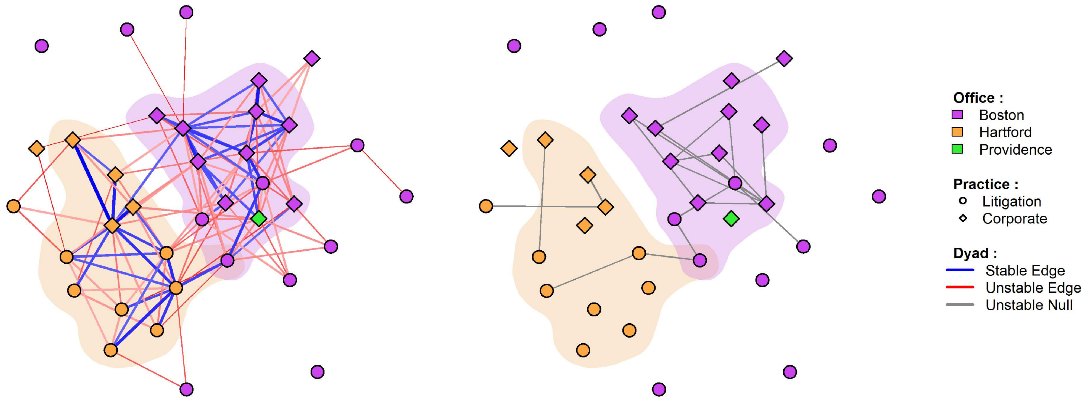


Figure 15: Illustration of the Lageza lawyer network with stable/unstable edges and nulls under the model. The edge stability is illustrated on the left panel. Blue and red are used to for stable and unstable edges, respectively. The line thinness of the stable edges are scaled with d_i. Unstable nulls are plotted on the right panel, with other nulls stable.

Although dyad stability can be quantified by calculating the change score for a dyad under a particular model without invoking the stable region, this notion of stability can be used in conjunction with the stable region to gain additional insight into further parameter optimization. This analysis provides a basis for predicting what changes in the network can be expected under the current social forces. Specifically, it points to the most vulnerable set of edges/nulls in the network. The external unstable edges between two groups suggest that cross-office ties tend to be somewhat fragile, and prone to disruption. The internal unstable edges connecting peripheral partners to core partners within groups likewise suggests an enhanced propensity towards turnover for collaborations involving these more marginal partners. By contrast, the unstable *nulls* among core actors suggest the potential for new collaborations among the most central partners, especially in the Boston office.

The local stability calculations identify the dyads that are most vulnerable to changes in the network, i.e., if changes *were* to happen, the unstable edges are the ones that would be expected to change the earliest. To see how this corresponds with an explicit dynamic process, we run 100,000 Metropolis trajectories until the first dyad toggle is accepted and record the number of times each dyad was toggled. We plot the fraction of each dyad toggle occurrence as a function of d_i for all dyad D_i. When $d_i < 0$ (left panel of fig.16), it indicates dyad D_i is stable and the probability of accepting an edge toggle is proportional to the ratio of target/alternative graph potentials. When $d_i > 0$ (right panel of fig.16), dyad D_i is unstable and the fraction of dyad toggles becomes flat and is equal to the probability of any dyad being sampled ($\binom{v}{2}^{-1}$, which is 1/630 in this case). This reflects the fact that the Metropolis acceptance probability becomes 1 when a move is favored. This experiment shows that the most vulnerable dyads are the ones that are outside of the stable region, where the probability of such a dyad toggle is equal to the probability of any dyad being sampled. When a dyad lies within the stable region, the probability of a toggle is an exponentially increasing function of negated distance to the closest hyperplane, maxed out at dyad sampling probability. This indicates that dyads lying close to the hyperplanes, although within the stable region, are also somewhat vulnerable to structural changes. It is also interesting to note that under this model there are more unstable edges than nulls, indicateing a tendency towards lowering network density by breaking established edges.

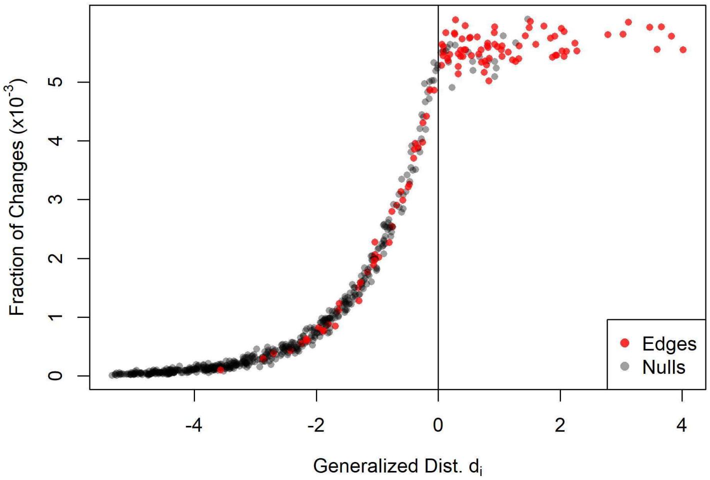


Figure 16: The dyad toggle occurrence fraction as a function of generalized distance d_i, recorded from 100,100 Metropolis trajectories until the first dyad toggle is accepted. The points on the left panel ($d_i < 0$) indicates dyads, w.r.t which the model is not stable, and points on the right ($d_i > 0$) indicate otherwise. Edges and nulls are differentiated by colors.

Finally, while we may consider stability with respect to externally defined groups (as above) we can also examine whether stability is associated with endogenously identified communities. Figure 17 shows stable and unstable edges within the Lazega network, with vertices labeled by community membership as identified via modularity minimization [32, 33]. We see that inferred community structure closely aligns with dyadic stability, with stable edges forming the respective cores of the two non-degenerate communities and ties between communities or to peripheral community members being predicted to be unstable. While the availability of covariate information here allows us to interpret the detected communities in this case as corresponding to the regional offices, such correspondence in other cases could be used to verify the robustness of inferred community structure: to the extent that the detection algorithm is identifying cohesive subgroups, we would expect to see a higher probability of stability for within-group versus between group edges. Alternatively, dyadic stability could also be employed as an input to community detection itself, with edge stability used e.g. to weight edge variables prior to segmentation. This may be particularly useful in networks with large numbers of ephemeral contacts for which direct measurement of tie strength is unavailable; even for very large networks in which ERGM parameter estimation by MCMC is not feasible, maximum pseudo-likelihood [34] or contrastive divergence [35] estimates may be sufficiently informative to improve algorithmic performance. This would seem to be a fruitful area for further investigation.

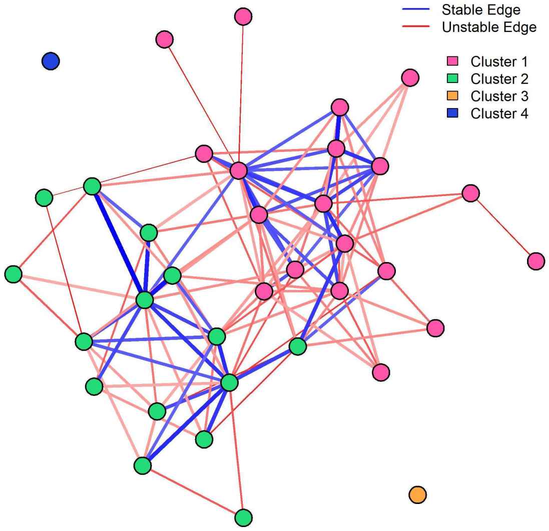


Figure 17: Dyadic stability by community structure. Clusters (vertex color) identified by modularity minimization are related to stability, with stable edges lying disproportionately within communities. Cross-cutting edges and edges to peripheral members are more often predicted to be unstable.

5 Network Models for Amyloid Fibril Formation

It has recently been demonstrated that ERGMs can serve as the basis for modeling protein aggregation events on both timescales and system size scales that are orders of magnitude beyond what is accessible using particle-based spatial modeling techniques [18, 36]. These approaches are particularly useful for modeling the spontaneous self-assembly of amyloid fibrils, which are filamentous aggregates of proteins characterized by locally ordered periodic structure. This periodicity is analogous to the periodic structure of crystals, the key difference being that crystal structures exhibit periodic structure along three axes of growth, while the periodicity of fibrils occurs along a single axis of growth (e.g. note the region of the network highlighted in Fig. 18 C). This periodic symmetry, that is a hallmark of fibril structure, poses unique challenges to model selection, as it exacerbates a well-known challenge in fitting ERGMs to observed network models via maximum pseudolikelihood estimation (MPLE) or other standard maximum likelihood optimization-based approaches: degeneracy [37]. It is in this realm of highly degenerate model spaces where using the stable region to pare down the parameter space to more tractable subspaces is particularly valuable. This approach is highlighted in Fig. 18, whereby the stable region is shown for a commonly observed fibril topology with respect to an ERGM with four sufficient statistics: $\theta \cdot t(g) = \theta_e edges + \theta_{2s} kstar(2) + \theta_{nsp1} nsp(1) + \theta_{nsp2} nsp(2)$. Specifically, 100,000 points were randomly selected from the surface of a sphere that is a constant edge parameter projection ($\theta_e = 109 - log(48)$) of a four dimensional hypersphere in the parameter space for the model. The parameters within the stable region were then used to carry out standard ERGM simulations. A graphlet-based approach [18, 36, 38, 39] was then used to quantify the fraction of nodes in the resulting graphs that belonged to subgraphs exhibiting the ladder-like local periodic structure characteristic of the fibril type known as a *2-ribbon*[18]. It is noteworthy that the stable region comprises a small subset of the entire parameter space, and that a substantial portion of that subspace is a fruitful region of parameter space where ERGM simulations produce high fibril yields. Since it was observed that simulations with high fibril yield were produced using parameters along the top edge of the stable region, 2030 additional simulations were run using parameters found just outside the stable region in the vicinity where the fibril-producing region of the stable region extended to the boundary (Fig. 19). Here it is shown that, although fibril producing parameters do exist just outside the locally stable region, the fibril-producing region does not extend far outside the stable region. These examples illustrate the utility of stable region-based approaches for model selection whereby the parameter search space is drastically pared down prior to parameter optimization. The fact that much but not all of the fibril-producing region is enclosed in the stable region in this application illustrates that the stable region can be particularly useful for informing the early stages of parameter optimization, but may not be ideal for strict regularization in the later stages of parameter optimization. Sample code for carrying out ERGM simulations using parameters drawn from various regions of the parameter space for this model can be found here.

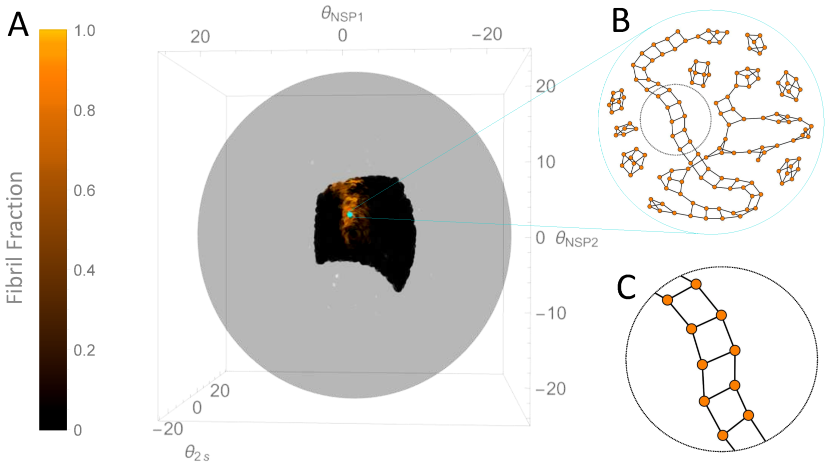


Figure 18: A. 100,000 randomly selected points from the surface of a sphere that is a constant edge parameter projection of the 4 dimensional hypersphere in parameter space for the ERGM family $\theta \cdot t(g) = \theta_e edges + \theta_{2s} kstar(2) + \theta_{nsp1} nsp(1) + \theta_{nsp2} nsp(2)$. Parameter values outside the stable region are shown as light gray dots, while the points inside the stable region are colored according to the fibril yield they produced under simulation. The point shown in cyan marks the parameters used to generate the simulation results shown in part B (also found in the SI section of Grazioli et al.). C. Detail of the characteristic 2-ribbon structure (a common amyloid fibril topology).

6 Discussion

In this paper, we proposed a novel approach to analyze graph stability, based on the relative favorability of a target graph vis-à-vis a set of alternative graphs under a probability model. The requirements for applicability of this approach are very general, being merely a target graph, a set of alternative graphs against which stability is to be assessed, and a model class that parameterizes network probability as a monotonically increasing function of the linear combination of parameter values and graph statistics ($\boldsymbol{\theta}^T \mathbf{t}(g)$). ERGMs are a widely used model class of such kind, with the probability of a graph being proportional to the exponentiated linear terms. The stable region, if it exists, is the region where the target graph is more probable than any of the proposed alternative graphs. We construct an $|S|$-by-K matrix M, each row of which is the generalized change score between the target graph and one of the alternative graphs (defined as $t(G') - t(G)$). Under this setting the stable set of a model is $\theta : \mathbf{M}\theta \in \mathbb{R}_-^{|S|}$. We show that the stable region (if it exists) is the interior of a convex K-polytope cone that points at the origin, and each facet of the cone is a row in M. We show an intuitive and easily implemented method to convert the facet representation to the ridge representation,

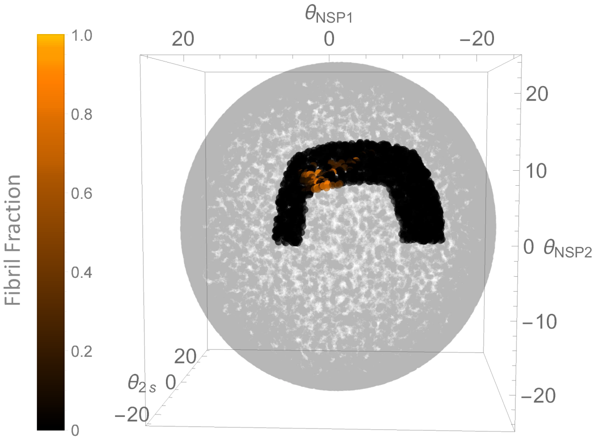


Figure 19: 120,000 points in parameter space randomly selected from the surface of the sphere shown in Fig. 18A, with ERGM simulations performed for points outside the stable region but near high-yield points on the stable region margin (color scaled according to fibril fraction). Although a small number of fibril-producing parameters do exist just outside the locally stable region (the orange points outside the top left corner of the stable region), they do not extend very far into the locally unstable region.

whose worst-case complexity is the same as the exhaustively searching through all possible intersections. Nevertheless, the best case or the average case complexity is a massive speedup over the exhaustive method.

Although this definition applies broadly to any target graph and arbitrary alternative set, certain alternative sets are particularly useful for providing intuition regarding model behavior and potential dynamics. For example, if we construct the alternative set to be all graphs that are Hamming distance one from the target graph, then stability against this set indicates that any one-step change is disfavored. For a random walk algorithm that assumes changes happen in series (i.e., one change at a time), this very local form of stability is an easily calculated approximation for dynamic stability (since any longer trajectory must still begin with a single step).

The ability to characterize stability for a given target across an entire model family also has potential applications for model screening and selection. For instance, given observed graph G_{obs} and family parameterized by $\mathbf{t}$, then the non-existence of a stable region for G_{obs} with respect to $\mathbf{t}$ indicates that there exists no model within the family for which G_{obs} is a locally unique mode; in settings for which G_{obs} is known to be stable, restricting attention to model families with non-empty stable regions may greatly speed model search. The lack of a stable region for G_{obs} also implies some degree of non-correspondence between the maximum likelihood estimator (MLE) of $\theta|G_{\text{obs}}$ and the observed graph. If $\hat{\theta} = \arg\max_{\theta} \Pr(G_{\text{obs}}|\theta)$ is the MLE of θ under family $\mathbf{t}$, then it is a standard property of exponential families that $\mathbf{E}_{\hat{\theta}} = \mathbf{t}(G_{\text{obs}})$. If the MLE exists, then it follows that the expected statistics of the corresponding model must agree with the observed data. However, as is also well-known [40], the mode of the distribution of statistics under the MLE need not be close to the observed value. While an empty stable set for G_{obs} under $\mathbf{t}$ does not determine whether this property holds, it obviously implies that the graph distribution under the MLE cannot have a locally unique mode at the observed graph. This may be useful for prospectively identifying model families for which additional care is needed in interpreting the MLE.

To demonstrate our approach we showed a simple social example - an isolated star graph - inspired by the structures arising in certain charismatic cult groups. Local stability ($S = H^{(1)}$) is particularly immediate in this case, as S includes only two isomorphism classes. Under a minimal model family, we verify that the stable region corresponds closely to the region that remains dynamically stable under random walk MCMC trajectories, with dynamic stability fading as one approaches the facets of the stable region. We further investigate the potential differences of graphs in S near the facet of the stable region and show how dynamic stability is related to the potential difference of the alternative graphs versus the target. When S is chosen to be $H^{(1)}$, any instability w.r.t. S can also be interpreted in terms of dyad vulnerability, that is, under the current model forces, if changes were to happen to the network, which dyad is most vulnerable to being toggled. We divide the parameter space into regions that represent the types of dyads that are most likely to undergo changes from both theoretical derivation and simulation.

We demonstrate a straightforward approach to exploring network stability under a given model, using the Lazega Lawyer dataset as an example. By simply constructing the $\mathbf{M}$ matrix and examining whether the vector $\mathbf{M}\hat{\boldsymbol{\theta}}$ is in the negative quadrant, i.e., $\mathbf{M}\hat{\boldsymbol{\theta}} \in \mathbb{R}_{-}^{|S|}$, we can assess whether the observed graph is predicted to be stable under the estimated model. When the graph is not locally stable, then we may further inquire into the stability of particular dyads (exploiting the relationship between dyad toggles and the elements of S). Thus, non-negative elements in $\mathbf{M}\hat{\boldsymbol{\theta}}$ suggest the instability of the corresponding dyads. This can aid in making predictions regarding potential future changes to the network under the current social forces (as parameterized by the estimated model). For the Lazega dataset, we are able to identify the structural characteristics of the network from dyad stability assessment. For example, by examining the stable edges we are able to identify two densely connected clusters centered at two of the larger offices; each exhibits a core-periphery structure. The unstable nulls are relatively concentrated in the Boston office, suggesting that the pressure to collaborate is greater in the Boston office, while unstable edges bridging offices suggest fragility in ties between units. Such insights may be useful for guiding additional empirical studies or modeling efforts.

Additionally, we demonstrated the utility of using our stable region-based approaches to pare down vast parameter spaces to much smaller stable subspaces by applying our methodologies to fitting ERGMs that can be used to model amyloid fibril producing protein aggregation events. For the 2-ribbon model studied here, the locally stable region of parameter space contains a large fraction of the region producing fibrillar topologies in equilibrium, demonstrating that the stable region can offer significant utility for initializing parameter searches.

The methodology introduced here offers a powerful new toolset for practitioners of network modeling. The techniques presented require no simulation, and are applicable to a wide range of problems. The one-step stability metric introduced herein is a straightforward implementation of the alternative set, and possesses an intuitive interpretation. At the same time, our method is amenable to any user-defined alternative set. Our approach also offers a quantitative tool for measuring instability in network structure due to either a strain imposed on a network by the social forces at play, or conversely, could indicate that the choice of social forces in the model could be poorly chosen. Finally, we also note that the formal correspondence between the ERGM form and Boltzmann distribution makes this approach useful in physical settings [18], where locally stable structures correspond to local energy minima in graph space. The ability to easily characterize the conditions under which particular graph structures are energetically favorable may be useful for studying the formation of complex materials, or the protein aggregates associated with Alzheimer's and other diseases [18].

References

[1] Jacob L. Moreno and Helen H. Jennings. "Statistics of Social Configurations". In: *Sociometry* 1.3/4 (1938), pp. 342–374.

[2] Duncan J. Watts. "The "New" Science of Networks". In: *Annu. Rev. Sociol.* 30 (2004), pp. 243–270.

[3] Scott A. Boorman. "A Combinatorial Optimization Model for Transmission of Job Information through Contact Networks". In: *Bell journal of Economics* 6.1 (1975), pp. 216–249.

[4] Peter Sheridan Dodds, Duncan J. Watts, and Charles F. Sabel. "Information Exchange and the Robustness of Organizational Networks". In: *Proceedings of the National Academy of Sciences* 100.21 (2003), pp. 12516–12521.

[5] Robin Cowan and Nicolas Jonard. "Network Structure and the Diffusion of Knowledge". In: *Journal of Economic Dynamics and Control* 28.8 (2004), pp. 1557–1575.

[6] David Krackhardt and R. N. Stern. "Informal Networks and Organizational Crises: An Experimental Simulation". In: *Social Psychology Quarterly* 51 (1988), pp. 123–140.

[7] Duncan S. Callaway, Mark E. J. Newman, Steven H. Strogatz, and Duncan J. Watts. "Network Robustness and Fragility : Percolation on random graphs". In: *Physical review letters* 85.25 (2000), p. 5468.

[8] Gunnar W. Klau and René Weiskircher. "Robustness and Resilience". In: *Network Analysis: Methodological Foundations*. Ed. by Ulrik Brandes and Thomas Erlebach. Berlin: Springer-Verlag, 2005. Chap. 15, pp. 417–437.

[9] Daron Acemoglu, Asuman Ozdaglar, and Alireza Tahbaz-Salehi. "Systemic Risk and Stability in Financial Networks". In: *American Economic Review* 105.2 (2015), pp. 564–608.

[10] Martina Morris and Mirjam Kretzschmar. "Concurrent Partnerships and the Spread of HIV". In: *Aids* 11.5 (1997), pp. 641–648.

[11] Martina Morris, Ann E. Kurth, Deven T. Hamilton, James Moody, and Steve Wakefield. "Concurrent Partnerships and HIV Prevalence Disparities by Race: Linking Science and Public Health Practice". In: *American Journal of Public Health* 99.6 (2009), pp. 1023–1031.

[12] Paul W. Holland and Samuel Leinhardt. "Transitivity in Structural Models of Small Groups". In: *Comparative Group Studies* 2 (1971), pp. 107–124.

[13] Ove Frank and David Strauss. "Markov Graphs". In: *Journal of the American Statistical Association* 81 (1986), pp. 832–842.

[14] Philippa E. Pattison and Garry L. Robins. "Neighborhood-Based Models for Social Networks". In: *Sociological Methodology* 32 (2002), pp. 301–337.

[15] Tom A. B. Snijders, Philippa E. Pattison, Garry L. Robins, and Mark S. Handcock. "New Specifications for Exponential Random Graph Models". In: *Sociological Methodology* 36 (2006), 99–154.

[16] Garry L. Robins, Philippa E. Pattison, and Jodie Woolcock. "Small and Other Worlds: Network Structures from Local Processes". In: *American Journal of Sociology* 110.4 (2005), pp. 894–936.

[17] Dean Lusher, Johan Koskinen, and Garry L. Robins, eds. *Exponential Random Graph Models for Social Networks: Theory, Methods, and Applications*. Structural Analysis in the Social Sciences. Cambridge: Cambridge University Press, 2012. DOI: 10.1017/CBO9780511894701.

[18] Gianmarc Grazioli, Yue Yu, Megha H. Unhelkar, Rachel W. Martin, and Carter T. Butts. "Network-based Classification and Modeling of Amyloid Fibrils". In: *Journal of Physical Chemistry, B* 123.26 (2019), pp. 5452–5462. DOI: 10.1021/acs.jpcb.9b03494.

[19] Gianmarc Grazioli, Rachel W. Martin, and Carter T. Butts. "Comparative Exploratory Analysis of Intrinsically Disordered Protein Dynamics using Machine Learning and Network Analytic Methods". In: *Frontiers in Molecular Biosciences, Biological Modeling and Simulation* 6.42 (2019). DOI: 10.3389/fmolb.2019.00042.

[20] Tom A. B. Snijders. "The Statistical Evaluation of Social Network Dynamics". In: *Sociological Methodology* 31 (2001), pp. 361–395.

[21] David R. Hunter, Pavel N. Krivitsky, and Michael Schweinberger. "Computational Statistical Methods for Social Network Analysis". In: *Journal of Computational and Graphical Statistics* 21 (4 2012), pp. 856–882.

[22] David Avis and Komei Fukuda. "A Pivoting Algorithm for Convex Hulls and Vertex Enumeration of Arrangements and Polyhedra". In: *Discrete & Computational Geometry* 8.3 (1992), pp. 295–313.

[23] C. Bradford Barber, David P. Dobkin, David P. Dobkin, and Hannu Huhdanpaa. "The Quickhull Algorithm for Convex Hulls". In: *ACM Transactions on Mathematical Software (TOMS)* 22.4 (1996), pp. 469–483.

[24] Noga Alon, Raphael Yuster, and Uri Zwick. “Finding and counting given length cycles”. In: *Algorithmica* 17.3 (1997), pp. 209–223.

[25] Doyle P. Johnson. “Dilemmas of Charismatic Leadership: The Case of the People’s Temple”. In: *Sociological Analysis* 40.4 (1979), pp. 315–323.

[26] Winston Davis. “Heaven’s Gate: A Study of Religious Obedience”. In: *Nova Religio: The Journal of Alternative and Emergent Religions* 3.2 (2000), pp. 241–267.

[27] David R. Hunter, Mark S. Handcock, Carter T. Butts, Steven M. Goodreau, and Martina Morris. “ergm: A Package to Fit, Simulate and Diagnose Exponential-Family Models for Networks”. In: *Journal of Statistical Software* 24.3 (2008).

[28] Mark S. Handcock, David R. Hunter, Carter T. Butts, Steven M. Goodreau, and Martina Morris. “statnet: Software Tools for the Representation, Visualization, Analysis and Simulation of Network Data”. In: *Journal of Statistical Software* 24.1 (2008), pp. 1–11.

[29] Emmanuel Lazega. *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership.* Oxford University Press on Demand, 2001.

[30] David R. Hunter. “Curved Exponential Family Models for Social Networks”. In: *Social networks* 29.2 (2007), pp. 216–230.

[31] Mark S. Handcock, David R. Hunter, Carter T. Butts, Steven M. Goodreau, Pavel N. Krivitsky, and Martina Morris. *ergm: Fit, Simulate and Diagnose Exponential-Family Models for Networks.* R package version 3.8.0. The Statnet Project (`https://statnet.org`). 2019. URL: `https://CRAN.R-project.org/package=ergm`.

[32] M. Girvan and M. E. J. Newman. “Community structure in social and biological networks”. In: *Proceedings of the National Academy of Sciences* 99.12 (2002), pp. 7821–7826. ISSN: 0027-8424. DOI: `10.1073/pnas.122653799`. eprint: `http://www.pnas.org/content/99/12/7821.full.pdf`. URL: `http://www.pnas.org/content/99/12/7821`.

[33] Mark E. J. Newman. “Modularity and Community Structure in Networks”. In: *Proceedings of the National Academy of Sciences* 103.23 (2006), pp. 8577–8582.

[34] David Strauss and Michael Ikeda. “Pseudolikelihood Estimation for Social Networks”. In: *Journal of the American Statistical Association* 85.409 (1990), pp. 204–212.

[35] P. N. Krivitsky. “Using contrastive divergence to seed Monte Carlo MLE for exponential-family random graph models”. In: *Computational Statistics and Data Analysis* 107 (2017), pp. 149–161.

[36] Yue Yu, Gianmarc Grazioli, Megha H Unhelkar, Rachel W Martin, and Carter T Butts. “Network Hamiltonian models reveal pathways to amyloid fibril formation”. In: *Scientific reports* 10.1 (2020), pp. 1–11.

[37] Sourav Chatterjee, Persi Diaconis, et al. “Estimating and understanding exponential random graph models”. In: *The Annals of Statistics* 41.5 (2013), pp. 2428–2461.

[38] Omer Nebil Yaveroglu, Sean M Fitzhugh, Maciej Kurant, Athina Markopoulou, Carter T Butts, and Natasa Przulj. “ergm. graphlets: A package for ERG modeling based on graphlet statistics”. In: *arXiv preprint arXiv:1405.7348* (2014).

[39] Tomaž Hočevar, Janez Demšar, et al. “Computation of graphlet orbits for nodes and edges in sparse graphs”. In: *Journ. Stat. Soft* 71 (2016).

[40] Tom A. B. Snijders. “Markov Chain Monte Carlo Estimation of Exponential Random Graph Models”. In: *Journal of Social Structure* 3.2 (2002).