A simple model of influence: Details and variants of dynamics

Colin Cooper*, Nan Kang, Tomasz Radzik[0000-0002-7776-5461], and Ngoc Vu

Department of Informatics, King's College London, UK
cooper.colin@gmail.com, nann.kang@gmail.com, tomasz.radzik@kcl.ac.uk, v.t.ngoc90@gmail.com

Abstract. We consider a simple model of establishing influence in a network. Vertices (people) split into influence groups and follow the opinion of the leader – the influencer – of their group. Groups can merge, based on interactions between influencers (the 'active vertices' of the network, while the followers are the 'passive vertices').
We study how the final number of influence groups depends on the way active vertices are chosen for interacting, considering two types of sparse graphs: the cycle C_n, which allows detailed analysis of various influencer algorithms, and random graphs $G(n, p)$ where $p = c/n$ for a constant c. We also introduce a simple dynamic Falling-Out model, which allows for rejection of opinion. In its most general form, as considered for $G(n, p)$, one of the two interacting influencers can decide to follow the other influencer, or they both can reject the opinion of the other influencer and instead choose other influencers to follow.
Our analysis for the cycle is based on solving systems of recurrences using generating functions, and our analysis for the random $G(n, p)$ graph uses the differential equation method.

Keywords: Random graphs and processes · Social networks and influence

1 Introduction

We study a simple model of influence in a network. In the model vertices (people) follow the opinion of the group they belong to. This opinion percolates down from an active (or opinionated) vertex, the *influencer*, at the head of the group. Groups can merge, based on edges between influencers (active vertices), so that the number of opinions is reduced. Eventually no active edges (edges between influencers) remain and the groups and their opinions become static.

Interest in models of social structure was promoted by the sociologist Robert Axelrod [2], who posed the question "If people tend to become more alike in their beliefs, attitudes, and behavior when they interact, why do not all such

* Research by C. Cooper supported at the University of Hamburg, DFG Project 491453517

differences eventually disappear?". This question was further studied by, for example, Flache *et al.* [8] and Moussaid *et al.* [12] who review various models of social interaction, generally based on some form of agency.

In our model, the emergence of separate groups occurs naturally due to lack of active edges between influencers. The exact composition of the groups and the opinion which influences them is a random outcome of the connectivity of the graph and the precise method of merging the groups. The paper [6] made an analysis of the Influencers model on an evolving version of the random graph $G(n, m)$. The graph has n vertices and m edges, which are selected randomly and added to the graph one at a time, in random order. If both end points of the current edge are active influencers, then one of them becomes a passive follower of the other, which remains as an active influencer. When no edges remain, the final influence structure is revealed.

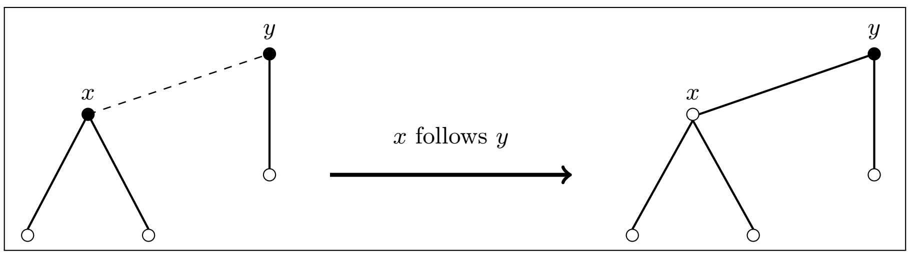

Fig. 1. Influencer x chooses a neighbouring influencer y and joins the group of y. Thus x becomes a follower of y along with the rest of its group.

In the current paper we study how the final number of influence groups depends on the way active vertices are chosen for interacting. We make this comparison for two types of sparse graphs: the cycle C_n, which allows detailed analysis of influencer algorithms; and random graphs $G(n, p)$ where each of the potential $\binom{n}{2}$ edges is present with probability $p = c/n$ for a constant c, independently of other edges.

We also introduce a simple dynamic model (Falling-Out) which allows for rejection of opinion. In its most general form, as considered for $G(n, p)$, one influencer can choose to follow another, or reject the opinion and instead follow another influencer. Although the dynamic is crude it does incorporate a natural component of human behaviour.

Joining Protocols. The Influencers problem is an instance of a general greedy process on a graph in which vertices are classified as active or passive. At any step, active vertices interact in some fashion, the result being that only one of the interacting vertices remains active, whilst the other interacting vertices become passive; the edges from the active vertex to the (now) passive ones being retained. This reduces the total number of active vertices until eventually there are no edges between them. At this point, the remaining active vertices become isolated and the process halts.

In this way, the process partitions the graph into disjoint subgraphs, which we call *fragments*, based on following the opinion of a neighbour. At any step, a fragment consists of a directed tree rooted at an active vertex, the current influencer, the edges pointing from follower vertices towards the root. This forms a simple model of influence where vertices in a fragment follow the opinion of the vertex they point to, and hence eventually that of the root.

The process is carried out on an underlying graph $G = (V, E)$. Initially all vertices of V are active, and all fragments are individual vertices. Let $A = A_t$ be the active set at step t and denote by $G[A]$ the subgraph of G induced by A. The simplest processes iterate the following steps.

1. *Vertex model*: Choose uniformly at random (u.a.r.) an active *non-isolated* vertex u and choose u.a.r. an active neighbour v of u. The edge uv is directed (u, v) and vertex u becomes passive (and follows the active vertex v).
2. *Edge model*: Choose a random edge from $G[A]$ and orient it u.a.r. If the oriented edge is (u, v), then u becomes passive (and follows the active vertex v).

Both processes end when $G[A]$ has no edges, and the final influencers are the final set A of remaining isolated active vertices.

Other processes studied here include variants of the edge model based on the active degrees of the edge endpoints and the following *Falling out (break edge) model*. Choose a random unexamined edge uv from $G[A]$. Vertices u and v then choose u.a.r. neighbours from $A \setminus \{u, v\}$, direct an edge to the chosen neighbours and become passive. If no such neighbour exists they become isolated.

Summary of previous results. The paper [6] made an analysis of the edge process on the random graph $G(n, m)$, choosing at step $1 \leqslant t \leqslant m \leqslant N \equiv \binom{n}{2}$ a random edge from the remaining $N - t + 1$ potential edges. Whenever both endpoints of the currently considered edge are active, one of them (selected u.a.r.) becomes passive. The results given in [6] include the following, among others.

- The number of fragments $a(m)$ in $G(n, m)$ is with high probability (w.h.p.) asymptotic to $F(m) = \frac{1}{1-(1-1/n)\sqrt{1-m/N}}$, for $m \ll N$, and this formula is an upper bound on the expected value of $a(m)$ for any $m \leqslant N(1 - o(1))$. The simulations indicate that this upper bound $F(m)$ may give the actual expected number of fragments for any $m \leqslant N(1 - o(1))$.
- The equivalent number of fragments in the presence of *stubborn vertices* (who accept followers, but refuse to follow). If $m = cN$, for a constant $c < 1$, then one stubborn vertex reduces the expected number of fragments to at most $\sqrt{1-c}\,F(m)$.
- The sizes of the fragments correspond to the lengths of the parts of a stick in the *stick breaking process* [11, 13]. The expected size of the largest of $k \geqslant 2$ remaining fragments is asymptotic to $(n/k) \log k$.

General remarks. As previously noted, the Influencers problem is an instance of a class of greedy processes on graphs, in which at each step some subset of active vertices interact and all but one of them become passive. For the Influencer problem the interactions are pairwise based on the existence of edges between active vertices. The selected edges are retained to form tree components rooted at currently active vertices.

For a given graph G let $\alpha(G)$ be the independence number, the size of the largest independent set. The final set S of active isolated vertices returned by the Influencer process on a graph G is an independent set. Thus $|S| \leqslant \alpha(G)$, and it is natural to compare the size of S with the sizes of sets computed by heuristics for large independent sets.

The algorithm Greedy-IS (Greedy Independent Set) chooses at each step an active vertex v, adds v to S (initially empty), and deletes v and all of its neighbours from the graph, continuing until there are no vertices left in the graph. Greedy-IS fits formally the general description of the Influencers process, and can be viewed as a variant in which a selected active vertex v makes all its active neighbours passive (followers of v). Greedy-IS is a well known attempt to solve a *maximization* problem, whereas there is no obvious objective function for Influencers. Thus, realistically, protocols like the vertex and edge models align better with the underlying idea of modeling emergence of influencers than Greedy-IS, which requires that the selected active vertex imposes its opinion on *all* its neighbours.

The following Lemma, quoted from [5], is classic and gives a bound for Greedy-IS for any graph G.

Lemma 1. *Greedy-IS outputs an independent set S such that $|S| \geqslant n/(\Delta+1)$ where Δ is maximum degree of G. This can be extended via Turan's theorem to prove $|S| \geqslant n/(d+1)$ where d is the average degree.*

Since $\alpha(G)$, the independence number of G, is an upper bound on the final number of influencers, it may be tempting to assume Greedy-IS also gives an upper bound for the Influencer process. For $G(n, c/n)$ when $c > 1$ this should be the case, see [1], as Greedy-IS is notoriously difficult to improve. For very sparse graphs such as a cycle (or $G(n, c/n)$, $c < 1$) this is not necessarily so. In Section 2 we compare a range of algorithms for the Influencer process on C_n, one of which returns larger (independent) sets than Greedy-IS. In Section 3 we give a more limited comparison for $G(n, p)$. In both types of graph we also consider the Falling-Out dynamic model which has no direct relationship to independence number.

We define the *influencer ratio*, which is the limiting expected fraction of active isolated vertices S. Let G_n be a graph or graph space parameterized by n, for example C_n or $G(n, p)$. For a given fragmentation algorithm F, let S_n be the set of isolated active vertices remaining after running F on (a random element of) G_n and define the influencer ratio as

$$\rho(F) = \rho(F, G_n) = \lim_{n\to\infty} \mathbb{E}\,|S_n|/n.$$

The probabilities here are taken over the randomness in F and the potential randomness of G_n. Thus for the maximum independent set problem, $\rho_{IS} = \rho_{IS}(G_n) = \lim_{n\to\infty} \mathbb{E}\,\alpha(G_n)/n$.

2 The Influencer Problem on the cycle C_n

As the independence number $\alpha(C_n) = \alpha(P_{n-1}) = \lfloor n/2 \rfloor$, we have $\rho(F) \leqslant \rho_{IS} = 1/2$ for any algorithm F applied to n-vertex cycle C_n or path P_n. The lower bound on the number of influencers is 1, achieved on P_n by the Left protocol, which keeps making the leftmost active vertex of the path passive; $\rho(\text{Left}) = 0$.

2.1 Results for the cycle C_n

Greedy-IS (G-IS) was solved for the path P_n by Flory [9], who showed that the expected greedy independence ratio $\rho(\text{G-IS})$ tends to $\zeta_2 = (1 - e^{-2})/2$ as the path length tends to infinity. The cycle C_n is asymptotically equivalent to P_n.

The simplicity of the cycle allows us to compare many related strategies for the Influencer model. The algorithms listed in Table 1 are all randomized, and the stated value of $\rho(A)$ is the limiting expected value ratio. In [14] N. Vu studied many of these variants of the influencer problem on the cycle C_n, as an example of graph fragmentation. In most cases the value $\rho(A)$ is obtained via generating functions, as the solution to a system of recurrences. These results also hold asymptotically for any class of n-vertex 2-regular graphs with at most $o(n)$ cycles of finite length. In particular they hold w.h.p. for the underlying graph of a random permutation, as the expected number of cycles length at most ℓ in a random permutation is $O(\log \ell)$.

1. **Active vertex, active neighbour.** A random active vertex is chosen and a u.a.r. active neighbour (if any).
   (a) **Vertex Passive.** The chosen vertex becomes passive (or isolated).
   (b) **Neighbour Passive.** The chosen neighbour becomes passive.
   (c) **Lower Passive.** The lower degree vertex becomes passive. Ties are broken in favour of the neighbour.
   (d) **Higher Passive.** The higher degree vertex becomes passive. Ties are broken in favour of the chooser.
2. **Active vertex, maximum-degree active neighbour.** An active vertex is chosen and a random active neighbour of maximum degree.
   (a) **Overall Max. Deg. Wins.** The vertex of maximum degree remains active. Ties are broken randomly.
   (b) **Chooser Wins.** The choosing vertex remains active.
   (c) **Max. Deg. Neighbour.** The maximum degree neighbour wins. Ties are broken randomly.

3. **High-Low.** The lowest degree neighbour of a random vertex of highest degree is made passive. In the cycle, as long as paths of length at least 3 remain, a vertex of degree 2 is chosen. If it is next to a path endpoint, the endpoint becomes passive, and otherwise a random neighbour becomes passive.
4. **Random Edge.** A random edge between active vertices is chosen and one end becomes passive.
5. **Falling-Out (Dynamic).** A random edge between active vertices is broken and, each endpoint attaches to its random active neighbour, if any.

Algorithm A	Influencer ratio $\rho(A)$	Value	Comments
Max. Indep. Set	$1/2$	0.5000	Upper bound
Greedy-IS	$\zeta_2 = (1 - e^{-2})/2$	0.4323	Flory [9]
Vertex Passive (VP)	$1/3$	0.3333	Section 2.2
Neighbour Passive	$1 - e^{-1/2}$	0.3935	
Lower Passive	$1 + e^{1/2} - \sqrt{2\pi}\ \mathrm{erfi}(1/\sqrt{2})$	0.2588	
Higher Passive	$1 - 1/2e - \sqrt{\pi/4}\ \mathrm{erf}(1)$	**0.4426**	Largest
Overall Max. Deg.	$\left[2(1 - e^{1/12}) - \int_0^1 (3x-2)e^{-x^3/6+x^2/4}\right] / e^{1/12}$	0.2959	
Chooser	$\left[1 - e^{1/6} - \int_0^1 (x-1)e^{-x^3/3+x^2/2}\right] / e^{1/6}$	0.2919	
Max. Deg. Nbr.	$1/3$	0.3333	Same as VP
High-Low	$\left[e^{5/6} - 1 - \int_0^1 xe^{-x^3/3+x^2/2}\right] / e^{5/6}$	**0.2366**	Smallest
Random Edge	$1/e$	0.3677	Section 2.2
Falling-Out	$\zeta_2 = (1 - e^{-2})/2$	0.4323	Dynamic

Table 1. Influencer algorithms on cycle C_n and their ratios.

The largest value, $\rho \sim 0.44256$, is from Higher Passive (higher degree endpoint becomes passive). This supports the view that picking a vertex of minimum degree improves the performance of Greedy-IS. Similarly, the smallest value, $\rho \sim 0.2366$, is from High-Low see [14], where a minimum degree neighbour of a vertex of maximum degree is made passive.

For C_n, the dynamic variant Falling-Out (called Burn-Edge in [14]), has the same influencer ratio ζ_2 as the influencer ratio of Greedy-IS determined by Flory [9]. To see the correspondence between these two processes, use numbers $0, 1, \ldots, n-1$ to label the consecutive vertices around the C_n' cycle for the Greedy-IS process and to label the consecutive edges around the C_n'' cycle for the Falling-Out process. Couple the processes by selecting in the current step a random active vertex i in C_n' in the Greedy-IS process and using the edge with the same label i in C''_n in the Falling-Out process. To see that this coupling works, check by induction that at the beginning of each step a vertex i is active in C'_n in the Greedy-IS process, if and only if, the edge with the same label i is active in C''_n in the Falling-Out process. The processes end at the same step (no active vertices in C'_n and no active edges in C''_n). The number of vertices in C'_n selected for the IS is the same as the number of broken edges in C''_n, which in turn is equal to the number of fragments in C''_n computed by the Falling-Out process. (When the Falling-Out process terminates on C''_n, there is exactly one broken edge between each pair of consecutive fragments.)

2.2 Analysis for the cycle C_n

In this section we give proofs for the Vertex Passive and Random Edge models. The Vertex Passive proof is short and simple and gives also the result for Maximum Degree Neighbour. The Random Edge proof is not given in [14]. For other cases the analysis is given in [14].

Vertex Passive algorithm (VP). The sampled active vertex follows its chosen active neighbour. Active vertices are sampled in a random order. This corresponds to a random permutation $(x_1, x_2, ..., x_n)$ of the vertex labels. When we process x_j, the ordered active set is $(x_{j+1}, ..., x_n)$. The vertices $x_1, ..., x_{j-1}$ are already either passive or isolated. To remove ambiguity, suppose vertex x_j follows the first occurrence of a neighbour (if any) in the sequence $(x_{j+1}, ..., x_n)$. If no such neighbour exists x_j is isolated.

Let y, z be the neighbours of a vertex x in C_n. The event that x is isolated occurs if y, z precede x in the random permutation. Of the six permutations of $\{x, y, z\}$, two put x last, so being isolated has probability $1/3$. Thus the expected number of active isolates is $n/3$.

Maximum Degree Neighbour (MDN). We explain why the expected number of fragments is the same as Vertex Passive (VP). On a cycle, VP and MDN act identically if the selected vertex has two active neighbours of the same degree, or only one active neighbour. Thus MDN can only differ from VP when a vertex next to a path endpoint is chosen, and the path is of length $L \geqslant 4$. Label the vertices $v_1, v_2, ..., v_L$. If vertex v_2 is chosen, in MDN it will attach to v_3 which has degree 2. The resulting active lengths of the sub-paths are 1 and $L-2$, with v_1, v_3 active. In VP, choosing v_2 in a path of length at least 3, corresponds to a case where v_2 is first in the permutation, and thus has two active neighbours. Thus v_2 will attach to either v_1 or v_3 depending which is next in the permutation order. In either case, the resulting active lengths are 1 and $L-2$.

Random Edge algorithm. At each step a random edge between active vertices is chosen and one end becomes passive. An active path of length L is a (maximal) path of L active vertices, and thus $L-1$ edges. A path of length one is an isolated active vertex. At step $t = 1$ a random edge of C_n is chosen and one endpoint becomes passive, resulting in an active path of length $n-1$ bordered by the passive vertex. As the algorithm proceeds, this path is broken into smaller active paths bordered at each end by passive vertices, until finally only isolated vertices (active paths of length 1) remain.

Let N_L be the expected number of isolated vertices obtained by applying the Random Edge algorithm to an active path of length L. Thus $N_1 = 1$, $N_2 = 1$, and we define $N_0 = 0$. Given a path of length at least three, with vertices $v_1, ..., v_L$ and edges $e_1, ..., e_{L-1}$, where $e_i = (v_i, v_{i+1})$. Let $B(i, L)$ be the expected number of isolated vertices resulting when we apply the algorithm to edge e_i. Then

$$B(1, L) = B(L-1, L) = \frac{1}{2}(N_1 + N_{L-2}) + \frac{1}{2}N_{L-1}$$

$$B(i, L) = \frac{1}{2}(N_{i-1} + N_{L-i}) + \frac{1}{2}(N_i + N_{L-i-1}), \quad 2 \leqslant i \leqslant L-1.$$

It can be seen that $B(i, L) = B(L-i, L)$ for all i. Thus for $L \geqslant 3$,

$$N_L = \sum_{i=1}^{L-1} B(i, L) = \frac{1}{L-1}\left(N_1 + N_{L-2} + N_{L-1} + \sum_{i=2}^{L-2}(N_{i-1} + N_i)\right)$$

$$= \frac{1}{L-1}\sum_{i=1}^{L-1}(N_{i-1} + N_i), \qquad \text{giving}$$

$$LN_L = N_L + \sum_{i=1}^{L-1}(N_{i-1} + N_i). \tag{1}$$

Observe that (1) holds also for $L = 2$ since $2N_2 = 2 = N_2 + (N_0 + N_1)$. Thus for $L \geqslant 3$, by subtracting from (1) the same formula for $L-1$, we get

$$LN_L - (L-1)N_{L-1} = N_L + N_{L-2}. \tag{2}$$

The expression (2) is also true for $L = 2$ and, by setting $N_{-1} = 0$, for $L = 1$. Let $G(x) = \sum_{L=1}^{\infty} N_L x^L$, multiply (2) by x^{L-1}, and sum up for $L \geqslant 1$ to obtain

$$G'(x) - xG'(x) = \frac{1}{x}G(x) + xG(x), \text{ which implies } G'(x) = \frac{(x + 1/x)}{1-x}G(x).$$

The general solution to this differential equation is

$$G(x) = A\frac{x}{(1-x)^2}e^{-x}.$$

As $G(0) = 0$, differentiate again to obtain $G'(0) = A = N_1 = 1$. To find the coefficient $[x^n]G(x)$ write

$$G(x) = \frac{x}{(1-x)^2}e^{-x} = \frac{x}{e(1-x)^2}e^{1-x}$$

$$= \frac{x}{e(1-x)^2}\left(1 + (1-x) + \frac{(1-x)^2}{2!} + \cdots + \frac{(1-x)^j}{j!} + \cdots\right)$$

$$= \frac{xe^{-1}}{(1-x)^2} + \frac{xe^{-1}}{(1-x)} + f(x),$$

where $f(x)$ is entire (defined for all real numbers) so $\lim_{n\to\infty}[x^n]f(x) = 0$. Thus

$$[x^n]G(x) = ne^{-1} + e^{-1} + o_n(1),$$

and

$$\rho = \lim_{n\to\infty} \frac{N_{n-1}}{n} = e^{-1}.$$

3 The Influencer Problem for random graphs $G(n,p)$

The following is known about the independence number of $G(n,p)$; see [10, Theorem 7.4]. Let $\varepsilon > 0$ be a fixed constant, then for $c \geqslant c(\varepsilon)$, w.h.p.

$$\left|\alpha(G(n,c/n)) - \frac{2n}{c}(\log c - \log\log c - \log 2 + 1)\right| \leqslant \frac{\varepsilon n}{c}.$$

If $p = c/n$ where $c \to \infty$, then $\mathbb{E}\,\alpha \sim 2n\frac{\log c}{c}n$ and thus $\rho_{IS} \sim 2\frac{\log c}{c}$, whereas for c constant this is not the case.

3.1 Results for $G(n,p)$ when $p = c/n$

We give results for the Influencer model in $G(n,p)$ when $p = c/n$ for a constant c, for the following algorithms.

1. **Vertex Passive.** An active vertex is chosen u.a.r. and follows a random active neighbour.
2. **Random Edge.** Pick a random active edge and make one endpoint passive.
3. **Falling Out. (Dynamic)** A random edge between active vertices is chosen and broken. Each endpoint attaches to a random active neighbour, if any.

General Falling Out model. We also consider the following generalisation of the Falling-Out model. With probability β, a random edge is broken and each endpoint vertex becomes passive by following another active neighbour, or isolated active, if no such neighbour exists. With probability $1-\beta$ the edge is retained; in which case one endpoint becomes passive while the other remains active. If $\beta = 0$, the process is the Random Edge model; if $\beta = 1$, it is the basic Falling Out model. The general case of $0 \leqslant \beta \leqslant 1$ interpolates between these two models.

Theorem 1. *In $G(n,p)$ with $p = c/n$ for a constant c, the Falling Out model with $0 \leqslant \beta \leqslant 1$ constant has the influencer ratio*

$$\rho_\beta = \frac{2(1+\beta) - 2\beta e^{-c}}{2 + c(1+\beta)}.$$

Algorithm A	Influencer ratio $\rho(A)$	$\lim_{c\to\infty}\rho(A)$	Comments
Maximum Indep. Set		$(2\log c)/c$	For c large
Greedy-IS	$(\log(c+1))/c$	$(\log c)/c$	
Vertex Passive	$(1-e^{-c})/c$	$1/c$	
Random Edge	$2/(c+2)$	$2/c$	
Falling-Out	$(2-e^{-c})/(c+1)$	$2/c$	Dynamic
Falling-Out(β)	$\dfrac{2(1+\beta)-2\beta e^{-c}}{2+c(1+\beta)}$	$2/c$	

Table 2. Influencers ratios for algorithms on $G(n,p)$, where $p=c/n$ for an arbitrarily large constant c.

3.2 Analysis for random graphs $G(n,p)$

Greedy-IS. For completeness, and as an introductory simple example of the methodology we use later, we sketch how the well-known ratio $\rho=\log(c+1)/c$ of Greedy-IS on $G(n,c/n)$ can be derived. Let A_t be the number of active vertices at step t. The size of the computed independent set is $s=\min\{t:|A_t|=0\}$. We have $A_0=n$ and

$$\mathbb{E}\left(A_{t+1}\mid A_t\right)=\begin{cases}A_t-1-p(A_t-1), & \text{if } A_t\geqslant 1,\\ 0, & \text{if } A_t=0.\end{cases}\tag{3}$$

were $p(A_t-1)$ is the expected number of neighbours of the selected vertex. Taking the expectation of both sides of (3) and rearranging, we get, where $q=1-p$,

$$\mathbb{E}\left(A_{t+1}\right)=q(\mathbb{E}\left(A_t\right)-1)+q\,\mathbb{P}(A_t=0).$$

This recurrence can be approximated (as long as $\mathbb{P}(A_t=0)=o(1)$) with the recurrence $a_0=n$, $a_{t+1}=q(a_t-1)$, and one can show that if $0\leqslant a_t=o(n)$, then $\mathbb{E}\left(A_t\right)=o(n)$ and $\mathbb{E}\left(s\right)=t+o(n)$. The recurrence for the sequence $(a_t)_{t\geqslant 0}$ solves to $a_t=(n+q/p)q^t-q/p$, giving $0\leqslant a_t=o(n)$ for $t\sim\rho n$, $\rho=\log(c+1)/c$.

Vertex Passive model. At each step, random active vertex is chosen and follows a random active neighbour (if any). Thus the number of active vertices decreases by one. Initially there are n active vertices. So after t steps there are $n-t$ active vertices.

Let $S(t)$ be the number of isolated influencers at the start of step t. Thus $S(0)=0$. As $S(t)$ increases by one if and only if the chosen vertex v has no active neighbours (no edges between v and any of the other $n-(t+1)$ active neighbours), we have

$$\mathbb{E}\,S(t+1)=S(t)+(1-p)^{n-(t+1)},$$

$$\mathbb{E}\,S(n)=\sum_{t=0}^{n-1}(1-p)^{n-t-1}=\frac{1-(1-p)^n}{1-(1-p)}.$$

Thus $\mathbb{E}\,S(n)\sim(1-e^{-np})/p$, and if $p=c/n$, $\rho\sim(1-e^{-c})/c$.

Other models. In Sections 3.3–3.5 we analyse the Random Edge and Falling-Out models. The analysis is related to the analysis of the random Greedy Matching presented in [7] and [10, Chapter 6.4]. Furthermore, the analysis is for the random $G(n,m)$ graph with $m = cn/2$ and we derive the influencer ratios ρ using the differential equation method. As any $m = cn/2 + o(n)$ gives the same ρ, and w.h.p. $G(n,p)$ for $p = c/n$ has $cn/2+o(n)$ edges, the influencer ratios for $G(n,p = c/n)$ are the same as for $G(n, m = cn/2)$. We note that to obtain ρ, we are only interested in expected values and not w.h.p. results.

3.3 Random Edge

The following notation borrows from [10, Chapter 6.4] and [4] which give a proof of a related problem. Let $G(t) = (A_t, E_t)$ denote the active random subgraph of $G(n,m)$ remaining after t iterations. Let $\nu(t) = |A_t| = n - t$ be the number of vertices and $\mu(t)$ the number of edges in $G(t)$. At each step t, we choose a random edge $e_t = \{x, y\}$, delete the vertex x from A_t, and all edges incident with x. For the sake of our analysis we reveal the random graph as we run the algorithm.

A priori we do not know the degrees of any of the vertices. We just know that at step t we have $\nu(t)$ vertices and a uniform random set of $\mu(t)$ edges. We reveal the location of one of these edges, which is equally likely to have any two distinct endpoints among the $\nu(t)$ active vertices. More specifically, at each step t we reveal the edge e_t by choosing a random pair of distinct vertices. After we reveal the location of that edge, we know that its two endpoints must have degree at least 1. But we only know that because we revealed the edge.

For each edge e' among the other $\mu(t) - 1$ edges we reveal whether or not e' shares an endpoint with x. Any e' meeting x is deleted. Let $d'_t(x)$ be the number of such edges. Conditional on the edge e_t and the (say) $d'_t(x) = k$ deleted edges, the remaining edges comprise a uniform random set of $\mu(t) - 1 - k$ edges on the remaining set of $\nu(t) - 1$ vertices. Thus we have

$$\mathbb{E}\left[\mu(t+1) \mid \mu(t)\right] = \mu(t) - 1 - \mathbb{E}\left(d'_t(x) \mid \mu(t), e_t = \{x, y\}\right). \tag{4}$$

There are $1 \cdot (\nu(t) - 2)$ arrangements for an edge with one end at x and the other at a vertex other than x, y. Thus

$$\begin{aligned}\mathbb{E}\left(d'_t(x) \mid \mu(t), e_t = \{x, y\}\right) &= (\mu(t) - 1) \cdot \frac{\nu(t) - 2}{\binom{\nu(t)}{2} - 1} = \frac{2(\mu(t) - 1)}{\nu(t) + 1} \\ &= \frac{2\mu(t)}{\nu(t)} + O\left(\frac{1}{n-t} + \frac{\mu(t)}{(n-t)^2}\right).\end{aligned} \tag{5}$$

Provided $t = dn$ for some constant $d < 1$, the error term on the RHS is $O(1/n)$.

From (4) and (5) we have

$$\mathbb{E}\left[\mu(t+1) \mid \mu(t)\right] = \mu(t) - 1 - \frac{2\mu(t)}{\nu(t)} + O\left(n^{-1}\right). \tag{6}$$

This leads us to consider the differential equation (DE) which will simulate the process w.h.p. Let $t = \tau n, M(\tau) = \mu(t)/n$, then

$$\frac{dM}{d\tau} = -1 - \frac{2M(\tau)}{1-\tau}, \qquad M(0) = \frac{c}{2}, \tag{7}$$

which has solution

$$M(\tau) = \frac{1}{2}(1-\tau)\left(c - \tau(c+2)\right).$$

The smallest positive root of $M(\tau) = 0$ is $\tau^* = c/(c+2)$, which gives

$$\rho = 1 - \tau^* = 2/(c+2). \tag{8}$$

It can be shown (see e.g. [4] or [10, Chapter 6.4]) that w.h.p. the process ends with an isolated active set of size $\tau^* n + o(n)$.

We only need expected values for the influencer ratio, and are using the DE method as a way to approximate the solution to non-standard recurrences; as we now explain. It can be checked that the solution $S(t) = nM(\tau)$ satisfies

$$S(t+1) = S(t) - 1 - \frac{2S(t)}{\nu(t)} + \frac{c+2}{n}.$$

Thus by the above and (6),

$$\mathbb{E}\,\mu(t+1) - S(t+1) = (\mathbb{E}\,\mu(t) - S(t))(1 - 1/(n-t)) + d_t/n.$$

Provided $t = \tau n$ for some constant $\tau < 1$, iterating this back to $S(0) = \mu(0) = cn/2$ the error term on the RHS is $O(t/n)$. So $\mathbb{E}\,\mu(t^*) = S(t^*) + O(1)$, and $\mathbb{E}\,\mu(t) = 0$ at some $t \sim t^*$.

3.4 Basic Falling-Out model

We continue using the ideas in the formulation above, giving a brief proof, and leaving aside details. When the random edge $e_t = \{x, y\}$ is exposed, then given $\mu(t)$ and $\nu(t)$, the random graph $G(t)$ is otherwise unknown. We delete both x and y, the edge $\{x, y\}$ and the $d'_t(x) + d'_t(y)$ remaining edges adjacent to x or y. Thus we have $\nu(t) = n - 2t$ and the analysis of remaining edges follows the random Greedy Matching algorithm, see [7] or [10, Chapter 6.4]. The recurrence

$$\mathbb{E}\left(\mu(t+1) \mid \mu(t)\right) = \mu(t) - 1 - \mathbb{E}\left(d'_t(x) + d'_t(y) \mid \mu(t), e_t = \{x, y\}\right)$$

gives rise to the following differential equation, analogous to (7),

$$\frac{dM}{d\tau} = -1 - \frac{4M(\tau)}{1-2\tau}, \qquad M(0) = \frac{c}{2}.$$

As before we used $\mathbb{E}\,d_t(x) = 1 + 2\mu/\nu + O(1/n)$, see (5). Hence

$$M(\tau) = \frac{1}{2}(1-2\tau)\left(c - 2(c+1)\tau\right). \tag{9}$$

Thus $M(\tau) = 0$ at $\tau^* = c/2(c+1)$; the expected proportional size of the final matching (the set of removed independent edges) is $c/2(c+1)$ (see [7]) and an expected proportion of $1 - 2\tau^* = 1/(c+1)$ remaining isolated vertices.

It remains to calculate the number of isolated vertices created when edges were being deleted. An isolated vertex was created whenever an endpoint of the selected edge did not have any other neighbours. Let $d'_t(x)$ be the other edges incident with x after edge $\{x, y\}$ was exposed. Using a balls-in-boxes model where we throw $2(\mu(t) - 1)$ edge endpoints into $\nu(t)$ boxes, the probability none of the remaining randomly allocated edge endpoints is incident with x is

$$\mathbb{P}(d'_t(x) = 0) = \left(1 - \frac{1}{\nu(t)}\right)^{2(\mu(t)-1)} \sim e^{-2\mu/\nu}.$$

Conditional on this a similar result holds for y.

Let $S(t)$ be the number isolated active vertices arising from edge deletion at step t, then given $\mu(t)$ and $\nu(t) = n - 2t$,

$$\mathbb{E}\, S(t+1) = S(t) + 2(1 + o(1))e^{-2\mu(t)/n-2t}, \qquad S(0) = 0.$$

Let $t = t/n,\ M(\tau) = \mu/n,\ \sigma(\tau) = S(t)/n$, then using (9),

$$\frac{d\sigma}{d\tau} = 2e^{-M(\tau)/(1-2\tau)} = 2e^{-c}e^{2(c+1)\tau}, \qquad \sigma(0) = 0.$$

This has solution

$$\sigma(t) = \frac{e^{-c}}{c+1}\left(e^{2(c+1)\tau} - 1\right).$$

At $\tau^* = c/2(c+1)$, $\sigma(\tau^*) = (1 - e^{-c})/(c+1)$. Combining this with the $1 - 2\tau^* = 1/(c+1)$ remaining fraction of isolated vertices, we obtain the influencers ratio ρ for the basic Falling-Out model as

$$\rho = \frac{2 - e^{-c}}{c+1}. \tag{10}$$

3.5 General Falling-Out model

For β constant, $0 \leqslant \beta \leqslant 1,\ \mathbb{E}\,\nu(t) = n - (1+\beta)t$. It follows as above that

$$\mathbb{E}\,\mu(t+1) \sim \mu(t) - 1 - \left(\frac{2\mu}{\nu} + \beta\frac{2\mu}{\nu}\right).$$

This gives the corresponding differential equation and its solution:

$$\frac{dM(\tau)}{d\tau} = -1 - \frac{2(1+\beta)M}{1-(1+\beta)\tau}, \qquad M(0) = c/2,$$

$$M(\tau) = \frac{1}{2}(1 - (1+\beta)\tau)(c - \tau(2 + c(1+\beta))).$$

Thus $M(\tau) = 0$ at

$$\tau^* = \frac{c}{2 + c(1+\beta)}. \tag{11}$$

The expected number of isolated (active) vertices arising from edge deletion is

$$\mathbb{E}\, S(t+1) \sim S(t) + 2\beta e^{-2\mu/\nu}.$$

Putting $\sigma(\tau) = S(t)/n,\ \tau = t/n,$ and $2\mu/\nu = c - \tau(2 + c(1+\beta))$ gives

$$\frac{d\sigma(\tau)}{d\tau} = 2\beta e^{-c} e^{\tau(2+c(1+\beta))}, \qquad \sigma(0) = 0,$$

which has solution

$$\sigma(\tau) = \frac{2\beta e^{-c}}{2 + c(1+\beta)} \left(e^{\tau(2+c(1+\beta))} - 1 \right).$$

It follows that

$$\sigma(\tau^*) = \frac{2\beta}{2 + c(1+\beta)} (1 - e^{-c}).$$

The expected fraction of isolated active vertices remaining at this point is $1 - (1+\beta)\tau^*$. Adding this to the above gives

$$\rho = \frac{2(1+\beta) - 2\beta e^{-c}}{2 + c(1+\beta)}.$$

In particular, $\beta = 0$ gives ρ for the Random Edge model as in (8), and $\beta = 1$ gives ρ for the Basic falling-Out model as in (10).

3.6 Formalizing the DE for w.h.p. results

The formulations above use the Differential Equation method as a device to estimated expected values; the method to do this being shown at the end of Section 3.3.

Our analysis uses the notation and methodology of [10, Chapter 6.4], which gives an exposition of random Greedy Matching in $G(n, m)$ for $m = cn$, c constant. (Note that our analysis is for $m = cn/2$.) Thus although we do not require it, our results hold w.h.p. and not just in expectation. The models we analysed are similar to Greedy Matching. We refer the reader to [10, Chapter 6.4] for details, noting that the conditions (P1)-(P4) and the definition of the domain D as given there differ by constants from our setting, while the event $\mathcal{E}$ that the maximum degree $\Delta(G) \leqslant \log n$ is identical. See also [3, 15] for further details.

4 Conclusions and further work

The simplicity of the cycle allows us to analyse a range of related algorithms for the Influencer process, a number of which we were able to generalize to $G(n, p)$ or

$G(n, m)$. It would be interesting to analyse more realistic variants of the Falling-Out model. For example an active vertex x picks an active neighbour y_1 and either follows y_1 or breaks the edge. In the latter case x picks another active neighbour y_2 and repeats this behaviour until they either find some neighbour y_k they agree with or become an isolated influencer. This may possibly be a more realistic model of social behaviour.

References

1. A. Coja-Oghlan and C. Efthymiou. On independent sets in random graphs, Random Structures and Algorithms, 47(3), 436-486, (2015).
2. R. Axelrod. The dissemination of culture: A model with local convergence and global polarization. Journal of Conflict Resolution, 41(2), 203–226, (1997).
3. P. Bennett and A. Dudek. A gentle introduction to the differential equation method and dynamic concentration, Discrete Mathematics, 345(12), (2022).
4. P. Bennett, C. Cooper and A. Frieze. Rainbow Greedy Matching Algorithms. `https://arxiv.org/abs/2307.00657`
5. Chandra Chekuri. University of Illinois Urbana-Champaign. Course notes. `https://courses.engr.illinois.edu/cs583/sp2018/Notes/packing.pdf`
6. C. Cooper, N. Kang, T. Radzik. A Simple Model of Influence, WAW 2023: Algorithms and Models for the Web Graph, 164-178 (2023).
7. M. Dyer, A.M. Frieze and B. Pittel, The average performance of the greedy matching algorithm, Annals of Applied Probability 3 (1993) 526-552.
8. A. Flache, M. Mäs, T. Feliciani, E. Chattoe-Brown, G. Deffuant, S. Huet and J. Lorenz. Models of Social Influence: Towards the Next Frontiers. JASSS, 20(4) 2, (2017). `http://jasss.soc.surrey.ac.uk/20/4/2.html`
9. P. J. Flory, Intramolecular reaction between neighboring substituents of vinyl polymers, Journal of the American Chemical Society, 61 6, 1518–1521 (1939)
10. A. M. Frieze and M. Karoński. Introduction to Random Graphs. CUP, (2016). Online version available from `https://www.math.cmu.edu/~af1p/Book.html`
11. L. Holst. On the lengths of the pieces of a stick broken at random. J. Appl. Prob., 17, pp 623-634, (1980).
12. M. Moussaïd, J. E. Kämmer, P. P. Analytis and H. Neth. Social Influence and the Collective Dynamics of Opinion Formation. PLOS ONE 8(11): e78433. (2013)
13. R. Pyke. Spacings, JRSS(B) 27:3, pp. 395-449 (1965).
14. N. Vu. Some approaches to graph fragmentation with application to clustering geo-tagged data. PhD Thesis, King's College London, (2018). `https://kclpure.kcl.ac.uk/portal/en/persons/ngoc-vu/studentTheses/`
15. N. Wormald. The differential equation method for random graph processes and greedy algorithms, In Lectures on Approximation and Randomized Algorithms, PWN, Warsaw, 73–155 (1999).